



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

## SLEDOVÁNÍ ŘIDIČE

DRIVER MONITORING

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

Bc. Matúš Pieger

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Miloslav Richter, Ph.D.

BRNO 2021

# Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

**Student:** Bc. Matuš Pieger

**ID:** 186162

**Ročník:** 2

**Akademický rok:** 2020/21

**NÁZEV TÉMATU:**

## Sledování řidiče

### POKYNY PRO VYPRACOVÁNÍ:

- 1) Nastudujte bezpečnostní systémy automobilu v návaznosti na řidiče.
- 2) Nastudujte metody zpracování obrazu vhodné pro detekování řidiče.
- 3) Navrhněte měřicí sestavu/pracoviště pro danou úlohu. Nasnímejte testovací snímky/sekvence pro různé vstupní situace.
- 4) Navrhněte a realizujte algoritmy vhodné pro zpracování vstupních dat. Zvolte vhodnou reprezentaci dat výstupních.
- 5) Proveďte vyhodnocení realizovaných algoritmů pomocí nasnímaných testovacích sad a proveďte zhodnocení navrženého řešení.

### DOPORUČENÁ LITERATURA:

Gonzalez R.C., Woods R.E.: Digital Image Processing, 4th edition, Pearson, 2017, ISBN 978-0133356724

**Termín zadání:** 8.2.2021

**Termín odevzdání:** 17.5.2021

**Vedoucí práce:** Ing. Miloslav Richter, Ph.D.

**doc. Ing. Petr Fiedler, Ph.D.**  
předseda rady studijního programu

### UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Tato diplomová práce se zabývá návrhem systémů pro sběr dat popisujících chování řidiče v automobilu. Tyto data slouží k detekci rizikového chování, kterého se řidič může dopouštět kvůli nepozornosti způsobené nižší nebo vyšší úrovní automatizace řízení vozidla. Práce nejprve popisuje existující bezpečnostní systémy, a to především v návaznosti na řidiče. Poté se zabývá návrhem potřebných měřicích scén a realizací nových systémů založených na zpracování vstupních obrazových snímků. Pro snímání je využita stereo kamera Intel RealSense D415. Jednotlivé systémy jsou otestovány v reálném prostředí vozidla. Na závěr práce obsahuje vyhodnocení úspěšnosti detekce vytvořených algoritmů, zamýšlí se nad nedostatky a možnými vylepšeními.

## **KLÍČOVÁ SLOVA**

automobilový průmysl, ADAS systémy, autonomní řízení, zpracování obrazu, stereo vidění, segmentace, detekce objektů, strojové učení

## **ABSTRACT**

This master's thesis deals with the design of systems for data collection which describe the driver's behaviour in a car. This data is used to detect risky behaviour that the driver may commit due to inattention caused by the use of either lower or higher levels of driving automation. The thesis first describes the existing safety systems, especially in relation to the driver. Then it deals with the design of the necessary measuring scenes and the implementation of new systems based on the processing of input images which are obtained via the Intel RealSense D415 stereo camera. Every system is tested in a real vehicle environment. In the end the thesis contains an evaluation regarding the detection reliability of the created algorithms, it considers their shortcomings and possible improvements.

## **KEY WORDS**

automotive, ADAS systems, autonomous driving, image processing, stereo vision, segmentation, object detection, machine learning

## **BIBLIOGRAFICKÁ CITACE**

PIEGER, Matúš. *Sledování řidiče*. Brno, 2021. Dostupné také z: <https://www.vutbr.cz/studenti/zav-prace/detail/134819>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Miloslav Richter.

## Prohlášení autora o původnosti díla

<b>Jméno a příjmení studenta:</b>	<i>Bc. Matúš Pieger</i>
<b>VUT ID studenta:</b>	<i>186162</i>
<b>Typ práce:</b>	<i>Diplomová práce</i>
<b>Akademický rok:</b>	<i>2020/21</i>
<b>Téma závěrečné práce:</b>	<i>Sledování řidiče</i>

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: **17. května 2021**

-----  
podpis autora

## PODĚKOVÁNÍ

Za přátelský přístup a dobré rady při vytváření práce chci poděkovat Ing. Miloslavovi Richtrovi Ph.D.

V Brně dne: **17. května 2021**

-----  
podpis autora

# OBSAH

<b>ÚVOD</b>	<b>12</b>
<b>1 AUTOMOBILOVÁ BEZPEČNOST A BEZPEČNOSTNÍ SYSTÉMY V AUTOMOBILECH</b>	<b>13</b>
1.1 Historie a vývoj ve 20. století	13
1.2 Současnost	15
<b>2 POKROČILÉ BEZPEČNOSTNÍ SYSTÉMY A AUTOMATIZACE ŘÍZENÍ</b>	<b>16</b>
2.1 Pokročilé asistenční systémy (ADAS)	16
2.2 Automatizace řízení	19
2.3 Stupně automatizace řízení	20
2.3.1 Level 0 – Bez automatizace	20
2.3.2 Level 1 – Asistence řidiče	20
2.3.3 Level 2 – Částečná automatizace	21
2.3.4 Level 3 – Podmíněná automatizace	21
2.3.5 Level 4 – Vysoká automatizace	21
2.3.6 Level 5 – Plná automatizace	21
2.4 Pokročilé asistenční systémy v návaznosti na řidiče	22
2.4.1 Systém rozeznávání opilého řidiče	23
2.4.2 Systém detekce ospalosti řidiče hlídáním jeho očí	24
2.5 Navrhované systémy kontroly řidiče	28
<b>3 MĚŘICÍ SESTAVA</b>	<b>29</b>
3.1 Volba typu kamery a možné problémy	29
3.2 Stereo kamera Intel RealSense	31
3.3 Návrh měřicích scén	32
3.3.1 Měřicí scéna č. 1 – Pohled na obličej a vrchní část těla řidiče	32
3.3.2 Měřicí scéna č. 2 – Pohled na volant řidiče	33
3.3.3 Měřicí scéna č. 3 – Pohled do prostoru pedálů	33
3.4 Nasnímání testovacích snímků	34
<b>4 NÁVRH A REALIZACE METOD DETEKCE ŘIDIČE A JEHO POHYBŮ</b>	<b>38</b>
4.1 Základní rozbor	38
4.2 Prvotní metody pro vstupní snímky z běžné kamery	41
4.2.1 Detekce metodou Gaussian mixture model	42

4.2.2	Detekce metodou Haarovy kaskády	46
4.2.3	Hranové operátory pro detekci mrkání	52
4.2.4	Zhodnocení navržených metod pro 2D snímky	53
4.3	Rozbor, návrh a realizace metod - vstupní snímky ze stereo kamery	54
4.3.1	Princip stereovidění	55
4.3.2	Funkce pro segmentaci na základě hloubky objektů v obrazu	57
4.3.3	Detekce náklonu řidiče do jednotlivých stran	60
4.3.4	Detekce držení volantu řidičem	72
4.3.5	Detekce připravenosti stlačit pedály	75
4.3.6	Detekce mrkání a zívání	77
4.3.7	Detekce telefonování	85
4.3.8	Detekce natočení hlavy do strany	88
4.3.9	Výsledný program a provázanost algoritmů	89
<b>5</b>	<b>ZHODNOCENÍ VYTVOŘENÝCH METOD PRO DETEKCI ŘIDIČE A JEHO POHYBŮ</b>	<b>92</b>
<b>6</b>	<b>ZÁVĚR</b>	<b>94</b>
	<b>BIBLIOGRAFIE</b>	<b>95</b>
	<b>SEZNAM PŘÍLOH</b>	<b>98</b>



# Seznam obrázků

Obrázek 1-1: Reklama na bezpečnostní pás vynalezen automobilkou Volvo	14
Obrázek 2-1: Ilustrace principu adaptivního tempomatu	17
Obrázek 2-2: Noční vidění a s tím spojená detekce chodců	17
Obrázek 2-3: Ilustrace principu systému varování před vybočením z jízdního pruhu	18
Obrázek 2-4: Ilustrace systému rozeznávání dopravních značek	18
Obrázek 2-5: Ukázka detekce chodců v běžném provozu	19
Obrázek 2-6: Vybrané ADAS systémy a jejich pole vnímání okolí vozidla	20
Obrázek 2-7: Grafický přehled jednotlivých úrovní automatizace řízení	22
Obrázek 2-8: Ilustrace analýzy alkoholu v dechu řidiče	24
Obrázek 2-9: Ilustrace analýzy alkoholu v krvi řidiče	24
Obrázek 2-10: Ukázka detekce bodů vymezujících plochu oka	25
Obrázek 2-11: Ukázka změny hodnoty EAR při mrknutí oka	25
Obrázek 2-12: Průběh korekcí volantu v čase - aktivní vs. ospalý řidič	26
Obrázek 2-13: Ukázka použití elektrod pro provedení EOG	27
Obrázek 3-1: Snímání bez a s nočním viděním	30
Obrázek 3-2: Stereo kamera Bumblebee 2 a Microsoft Kinect	30
Obrázek 3-3: Stereo kamera Intel RealSense	31
Obrázek 3-4: Návrh měřicí scény typu pohled na obličej řidiče	33
Obrázek 3-5: Návrh měřicí scény typu pohled na volant	33
Obrázek 3-6: Návrh měřicí scény typu pohled do prostoru pedálů	34
Obrázek 3-7: Poloha kamery na malém stativu pro snímání měřicí scény č. 1	35
Obrázek 3-8: Měřicí scéna č. 1	35
Obrázek 3-9: Poloha kamery na velkém stativu pro snímání měřicí scény č. 2	36
Obrázek 3-10: Měřicí scéna č. 2	36
Obrázek 3-11: Poloha kamery na velkém stativu pro snímání měřicí scény č. 3	37
Obrázek 3-12: Měřicí scéna č. 3	37
Obrázek 4-1: Analýza měřicí scény č. 1 v kontextu navrhovaných systémů	39
Obrázek 4-2: Analýza měřicí scény č. 2 – systém detekce držení volantu	40

Obrázek 4-3: Analýza měřicí scény č. 3 – systém detekce připravenosti stlačit pedály	40
Obrázek 4-4: Ukázka jedné iterace GMM algoritmu na datech výšky skupiny lidí	43
Obrázek 4-5: Rozdělení shluků za použití různého množství Gaussových funkcí	44
Obrázek 4-6: Rozdělení vstupního obrázku na jednotlivé shluky algoritmem GMM	44
Obrázek 4-7: Provedení morfologického uzavření na shluk představující hlavu	44
Obrázek 4-8: Výsledná detekce hlavy algoritmem GMM	45
Obrázek 4-9: Algoritmus GMM pro segmentaci rukou	46
Obrázek 4-10: Možné použití algoritmu GMM pro detekci rukou na volantu	46
Obrázek 4-11: Vybrané typy Haarových příznaků	47
Obrázek 4-12: Převedení vstupního obrázku na integrální obrázek	48
Obrázek 4-13: Kaskádní klasifikátor	50
Obrázek 4-14: Detekce obličeje metodou Haarovy kaskády	51
Obrázek 4-15: Detekce obličeje metodou Haarovy kaskády – pohled dolu	51
Obrázek 4-16: Detekce očí metodou Haarovy kaskády	52
Obrázek 4-17: Plně otevřené oko řidiče vs. plně zavřené oko řidiče	52
Obrázek 4-18: Obraz hran u plně otevřeného vs. plně zavřeného oka	53
Obrázek 4-19: Obraz hran u plně otevřeného vs. plně zavřeného oka u metody GMM	53
Obrázek 4-20: Model stereo kamery	55
Obrázek 4-21: Epipolární geometrie a zjednodušení problému korespondence	56
Obrázek 4-22: Základní blokové schéma programu	58
Obrázek 4-23: Testování segmentace na základě hloubky v obraze	59
Obrázek 4-24: Stanovení referenční polohy hlavy pro měření	61
Obrázek 4-25: Vyznačení obálky hlavy ve snímku s odděleným pozadím	62
Obrázek 4-26: Krajní body pro detekci náklonu do stran	63
Obrázek 4-27: Změna polohy krajních bodů při náklonu doleva a doprava	64
Obrázek 4-28: Detekce ramen při nadzvednutí	65
Obrázek 4-29: Problematické zvednutí ruky	66
Obrázek 4-30: Problematické zvednutí ruky - plocha	67
Obrázek 4-31: Bezpečné okno	67
Obrázek 4-32: Testování náklonu hlavy do stran	69
Obrázek 4-33: Testování náklonu hlavy dopředu	70

Obrázek 4-34: Testování odolnosti vůči falešným detekcím	71
Obrázek 4-35: Ukázka chybně vyhodnocené situace při zvednutí ruky	71
Obrázek 4-36: Oddělení pozadí u měřicí scény č. 2	72
Obrázek 4-37: Ukázka segmentovaných ploch u měřicí scény č. 2	73
Obrázek 4-38: Limitní situace při držení volantů jednou rukou	74
Obrázek 4-39: Testování detekce držení volantů	74
Obrázek 4-40: Oddělení pozadí u měřicí scény č. 3	75
Obrázek 4-41: Ukázka segmentovaných ploch u měřicí scény č. 3	76
Obrázek 4-42: Limitní situace položení nohy k pedálům	76
Obrázek 4-43: Testování detekce připravenosti řidiče stlačit pedály	77
Obrázek 4-44: Výpočet gradientů v analyzovaném snímku	79
Obrázek 4-45: Příklad výpočtu histogramu orientovaných gradientů	80
Obrázek 4-46: Charakteristické body obličeje prediktoru z knihovny dlib	81
Obrázek 4-47: Testování prediktoru tvaru obličeje	82
Obrázek 4-48: Testování prediktoru tvaru obličeje – řidič s brýlemi	82
Obrázek 4-49: Ukázka změny nalezených ploch při zavření očí a při zívání	83
Obrázek 4-50: Testování detekce zívání	84
Obrázek 4-51: Testování detekce usínání, resp. pozornosti řidiče	84
Obrázek 4-52: Ukázka zvětšení měřené plochy pro detekci telefonování	85
Obrázek 4-53: Možné úchopy při telefonování	87
Obrázek 4-54: Testování detekce telefonování	88
Obrázek 4-55: Testování algoritmu pro detekci natočení hlavy do stran	89
Obrázek 4-56: Ukázka menu pro zvolení potřebných konstant	90
Obrázek 4-57: Vývojový diagram – měřicí scéna č. 1	91
Obrázek 4-58: Vývojový diagram – měřicí scéna č. 2 a 3	91

## ÚVOD

21. století se sebou přináší pořád větší a větší možnosti využití počítačových technologií. Tento vývoj se nevyhýbá ani automobilovému průmyslu. V minulosti spočíval vývoj automobilů především ve správném návrhu a konstrukci mechanických částí vozidla. Dnes se k této oblasti vývoje ve velké míře přidružuje vývoj softwaru. Může se jednat o řídicí software, který se nachází např. v řídicí jednotce motoru. Dále se může jednat o software, který vytváří tzv. infotainment, neboli informace zobrazující se na palubním počítači vozidla. Tato práce se ovšem bude zabývat softwarem určeným pro zvýšení míry automatizace a s tím související bezpečnosti, jak řidiče, tak i všech ostatních účastníků provozu.

Software, který tvoří tyto systémy, patří do množiny pokročilých asistenčních – bezpečnostních systémů. Tyto systémy jsou částečně, případně úplně schopny nahradit řidiče. Jejich výhodou je, že jsou schopny na nebezpečné situace reagovat mnohem rychleji, než člověk, jsou neunavitelné a pořád k dispozici. Díky tomu je možné významně snížit riziko dopravních nehod. Na druhou stranu to ale může vést k tomu, že řidič ztrácí koncentraci a nevěnuje se řízení. Podle toho, jak moc musí řidič zasahovat do řízení vozidla, rozeznáváme 6 kategorií automatizace, které budou rozebrány v kapitole *Stupně automatizace řízení*.

Cílem diplomové práce bude navrhnout systémy pro sběr dat, které budou kontrolovat chování, resp. pozornost řidiče a jeho připravenost převzít řízení v případě neočekávané situace v provozu, se kterou si asistenční systémy nebudou vědět poradit. Míra jeho nutné pozornosti bude záviset od stupně automatizace, kterou bude jeho vozidlo obsahovat.

Navržené systémy budou využívat principů zpracování obrazu. To znamená, že vstupem pro vyhodnocování pozornosti budou sekvence snímků řidiče. Na základě jejich softwarového vyhodnocování se následně určí míra pozornosti a připravenosti převzít řízení.

Diplomová práce bude rozdělena na několik částí. První kapitola se bude obecně zabývat bezpečností v automobilovém průmyslu, a to jak historií, vývojem, až po současnost. Ve druhé části se čtenář dozví o jednotlivých pokročilých asistenčních systémech, o principech automatizace v řízení a problémech, které zde mohou nastat. Také zde najde přehled způsobů, jak lze v současnosti hlídat pozornost řidiče. Nakonec zde bude proveden teoretický návrh systémů, které si diplomová práce dává za cíl vytvořit. V další části bude proveden rozbor měřicí sestavy, konkrétněji se bude jednat o výběr vhodné kamery pro snímání vstupních snímků a také o rozbor typů jednotlivých měřicích scén, které bude nutné nasnímat. Čtvrtá a pátá kapitola budou obsahovat podrobný popis samotné realizace a vyhodnocení navržených algoritmů.

# 1 AUTOMOBILOVÁ BEZPEČNOST A BEZPEČNOSTNÍ SYSTÉMY V AUTOMOBILECH

V této kapitole bude rozebrán obecný náhled na bezpečnost a bezpečnostní systémy v automobilech především z historického pohledu a z pohledu vývoje až do současnosti.

Jízda, resp. cestování automobily, patří mezi nedílnou součást našich životů. Automobily využíváme pro přesun do práce, do školy, na nákup, k lékaři nebo třeba pro cestu na dovolenou. Odhaduje se, že je v současnosti v provozu zhruba 1 miliarda vozidel. Zároveň platí, že každá jízda automobilem je z důvodu rychlosti, kterou je vozidlo schopné vyvinout, potenciálně nebezpečná. Riziko zranění hrozí jak při srážce dvou vozidel navzájem, tak i při nárazu vozidla do jakékoliv pevné překážky. Automobily jsou potenciálně nebezpečné i pro chodce nebo cyklisty.

Nejnovější data říkají, že jenom v České republice zemře průměrně 550 lidí ročně v důsledku dopravních nehod [1]. Ve světě se tento počet odhaduje na přibližně 1,35 miliónů zemřelých ročně. Podle výzkumu americké organizace NHTSA, spadající pod ministerstvo dopravy USA, je v drtivé většině (94 %) na vině člověk a jeho chybovost, pouze u 2 % nehod lze říct, že byly způsobeny závadou vozidla a další 2 % lze přisoudit špatným povětrnostním podmínkám [2]. Je tedy zřejmé, že otázka automobilové bezpečnosti hraje velkou roli a jejím zvyšováním jsme schopni zachránit spoustu životů. Zatímco v minulém století se vývoj soustředil hlavně na systémy zabezpečující pasivní ochranu posádky v případě nehody, v současnosti vývoj směřuje především k aktivní bezpečnosti. Systémy aktivní bezpečnosti jsou právě systémy, které jsou schopny eliminovat chyby řidiče. Ty jsou, jak bylo zmíněno, největším faktorem při nehodovosti vozidel. V kapitolách níže bude proveden stručný přehled historie vývoje systémů podílejících se na zvyšování automobilové bezpečnosti od samotného počátku až po současnost.

## 1.1 Historie a vývoj ve 20. století

Již od dob, kdy vznikl první kočár schopný pohybu bez využití koňské síly, hledal člověk způsoby, jak snížit nebezpečí jízdy ve velkém, těžkém, plechovém objektu. V úplných počátcích se jednalo o nebezpečí vylekaných koní, jízdy na bahnité silnici nebo převrácení. Poté co se počet těchto prvních automobilů zvyšoval, objevil se problém narážení jeden do druhého [3].

Automobilová bezpečnost byla na vzestupu od prvního dne vzniku automobilu. Zatímco v roce 1913 byl podíl úmrtnosti na 10 tisíc vozidel roven 33,38 osob, podle

nejnovějších dat z roku 2018 je v současnosti podíl pouze 1,42 osoby [4]. Jedná se o znatelný rozdíl, pořád je zde ovšem velký prostor ke zlepšení.

Prvních 50 let od vzniku prvního automobilu přineslo spoustu bezpečnostních zlepšení, která riziko úmrtí výrazně snížila. Již v roce 1903 byly vynalezeny stěrače, pár let poté následovala zpětná zrcátka, objevila se také světla, která umožnila jízdu po západu slunce. V tomto období také vznikly směrovky ve formě ramen, která se zvedala na základě stisknutí elektrického tlačítka. V roce 1921 se v autech objevily hlavové opěrky, které snižovaly riziko zranění při zadním nárazu do vozidla. Šest let poté bylo poprvé pro čelní okno použito laminované sklo, které mělo zabránit jeho rozpadu a zranění posádky při čelním nárazu. Již v roce 1934 byl proveden první crash-test, proces, který je dnes naprostou samozřejmostí při vývoji automobilů.

Všechny zmíněné vynálezy z období mezi léty 1900 až 1950 jsou dodnes neodmyslitelnou součástí každého automobilu. Asi nejvýznamnější bezpečnostní zlepšení ovšem přinesly 50. léta 20. století. Jednalo se o vynález a rozšíření využití bezpečnostního pásu. Bezpečnostní pás je dodnes jednou z nejefektivnějších bezpečnostních pomůcek, jelikož významně snižuje riziko nárazu samotné osoby do pevné překážky při nárazu vozidla. Jeho povinnost nošení se ale zavedla až o několik desítek let později, což se z dnešního pohledu jeví jako nepochopitelné. Spolu s bezpečnostními pásy se v tomto období začaly také využívat airbagy, které rovněž nemalou mírou přispěly k ochraně posádky při nárazech. Kombinace těchto bezpečnostních prvků posunula automobilovou bezpečnost o mílové kroky dopředu.



Obrázek 1-1: Reklama na bezpečnostní pás vynalezen automobilkou Volvo [5]

Zatímco výše vyjmenované vynálezy byly založeny především na mechanických principech, s nástupem 70. a 80. let 20. století se ke slovu začalo dostávat využití elektroniky. V tomto období se v automobilech začal objevovat systém kontroly nezapnutého pásu a také systém ABS. Jedná se o systém, který zabraňuje zablokování kol při brzdění a následné ztrátě adheze mezi koly a vozovkou. Tím umožňuje zachování stability a ovládání vozidla například při prudkém brzdění. Skutečný rozmach elektroniky ve vozidlech ovšem nastal až v 90. letech, kdy se ve vozidlech poprvé objevily zadní

parkovací kamery a další jízdní asistenty. Příkladem jsou systémy jako ESP – elektronický stabilizační program, který rozšiřuje funkci ABS nebo BAS, jenž monitoruje rychlost a intenzitu sešlápnutí brzdového pedálu. Na základě toho je schopen rozpoznat kritickou situaci a zvýšit tlak v brzdě soustavě pro dosažení větší brzdě síly. V 90. letech také pokračovalo zlepšování soustavy airbagů. V roce 1997 byla vytvořena organizace EURO NCAP, která je do dnešní doby největší organizací zabývající se nárazovými zkouškami automobilů [6].

## **1.2 Současnost**

S příchodem nového milénia, až dodnes, vývoj bezpečnostních systémů nabíral na intenzitě. Toto zrychlování vývoje souviselo především se zrychlováním nárůstu výpočetního výkonu. V rámci této doby se začíná pořád více mluvit o tzv. pokročilých asistenčních systémech (ADAS). Jedná se o systémy založené na jiném principu oproti starším systémům. Hlavním rozdílem je, že zatímco starší bezpečnostní systémy patřily do kategorie pasivních bezpečnostních opatření, což znamená, že byly navrženy pro minimalizaci zranění při nehodě, ADAS systémy se více zaměřují na prevenci nehod a považujeme je za systémy aktivní. Množina ADAS systémů je poměrně rozsáhlá a bude jim věnována následující kapitola.

## 2 POKROČILÉ BEZPEČNOSTNÍ SYSTÉMY A AUTOMATIZACE ŘÍZENÍ

Tato kapitola se věnuje skupině pokročilých asistenčních systémů s důrazem na systémy s přímou návazností na řidiče. Je zde také probrána souvislost s automatizací řízení.

### 2.1 Pokročilé asistenční systémy (ADAS)

Jak bylo již zmíněno, pokročilé asistenční systémy fungují na rozdíl od pasivních systémů, jako prevence proti nehodám. Jsou to systémy, které spoléhají na data, popisující vnější prostředí automobilu. To je rovněž odlišuje od běžných bezpečnostních systémů.

Zdroje dat mohou být různé. Může se jednat o různé snímače, radary, nebo kamery s následným zpracováním obrazu. Do této kategorie lze zařadit i starší systémy, jako ABS nebo ESP, které využívají informace ze snímačů. Novější ADAS systémy ovšem obvykle používají data z více zdrojů najednou.

Pokročilé asistenční systémy je možné rozdělit podle kritéria návaznosti na řidiče. Některé z ADAS systémů pracují na zvýšení bezpečnosti a prevenci nehod bez ohledu na řidiče, jiné slouží přímo pro kontrolu chování a pozornosti řidiče a varování v případě, že se řidič nevěnuje řízení. Tyto systémy jsou integrální součástí snah o automatizaci řízení, které se věnuje kapitola se stejným jménem.

Pro lepší pochopení souvislostí mezi klasickými ADAS systémy, ADAS systémy s návazností na řidiče a s nimi spojenou automatizací řízení je níže uveden přehled nejvýznamnějších ADAS systémů za posledních 20 let.

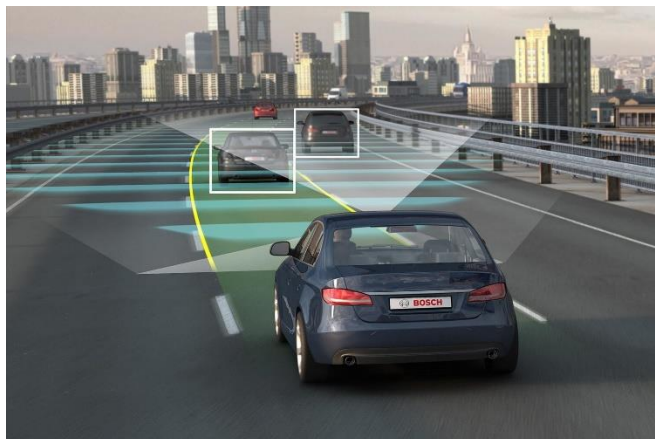
#### **Adaptivní tempomat:**

Vznik prvního adaptivního tempomatu se datuje do roku 1992, většího rozšíření se ale tento systém dočkal až kolem roku 2000.

Adaptivní tempomat je schopen udržovat zvolenou rychlost vozidla a zároveň pomocí automatického brzdění a přidávání udržovat bezpečnou vzdálenost mezi vozidlem a jiným vozidlem před ním. Jako zdroj dat obvykle využívá radar, případně laser zabudovaný v přední masce vozidla, který konstantně skenuje okolí vozidla. Účinnost se snižuje při špatném počasí (déšť, sníh), nebo při znečištění radarové jednotky.

Tento systém funguje jako pomůcka pro řidiče při řízení především na dálnici nebo při mimoměstské jízdě.





Obrázek 2-1: Ilustrace principu adaptivního tempomatu [7]

### Noční vidění:

Funkce nočního vidění je založena na principu termovizních kamer, které vytváří snímky na základě infračerveného vyzařování snímaných objektů. Jedná se o tzv. pasivní systém nočního vidění. U aktivního systému zároveň dochází k osvětlování silnice infračerveným světlem. Dosah vidění je u těchto systémů až do 300 m. Díky tomu je řidič lépe schopen vnímat okolí a také se mu zvětšuje dohledová vzdálenost ve tmě nebo ve špatných povětrnostních podmínkách. Jako zobrazovací médium slouží displej buď v rámci palubního počítače, nebo přímo na přístrojovém panelu.

Technologie byla poprvé představena v roce 2000, dodnes je ale možné ji najít především u prémiových vozidel.



Obrázek 2-2: Noční vidění a s tím spojená detekce chodců [8]

### Systém varování před vybočením z jízdního pruhu:

Další ze systémů z prvního roku nového tisíciletí, který se na trhu objevil, byl systém varování před vybočením z jízdního pruhu (LDWS). Tento systém využívá hned tři zdroje dat popisující vnější prostředí automobilu. Jedná se o kamery za čelním sklem, obvykle namontované vedle zpětných zrcátek, laserové snímače v přední části vozidla

a infračervené snímače. Systém využívá zpracování obrazu třeba ve formě hranových detektorů a tyto data vyhodnocuje v reálném čase. V případě detekce vybočení z jízdního pruhu je řidič upozorněn zvukovou výstrahou nebo vibracemi volantů. Více pokročilé LDWS systémy jsou v případě, že řidič nezareaguje, schopné převzít řízení.



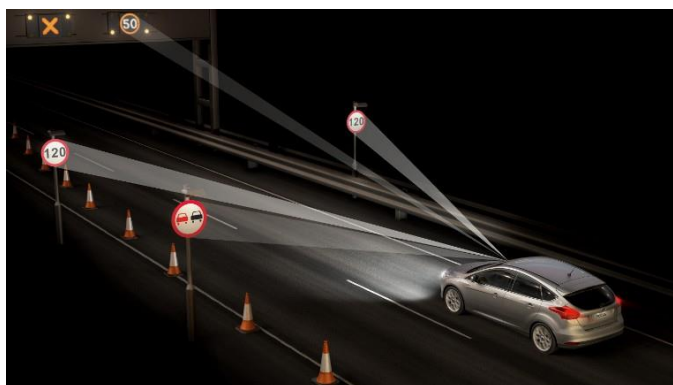
Obrázek 2-3: Ilustrace principu systému varování před vybočením z jízdního pruhu [9]

### **Systém hlídání mrtvého úhlu:**

Tento systém byl poprvé představen v roce 2004. Jeho podstatou je hlídání prostoru vedle a za vozidlem. Cílem je varovat řidiče v případě, že chce změnit jízdní pruh a v jeho mrtvém úhlu se zrovna nachází jiný automobil. Varování může být zobrazeno přímo na zpětném zrcátku nebo se použije zvuková výstraha. Pro detekci jiných vozidel se obvykle používají ultrazvukové snímače.

### **Systém rozeznávání dopravních značek:**

V roce 2009 byl na trh uveden systém pro detekci dopravních značek. Jedná se o software, který využívá data z přední kamery, snímající okolí vozu. Detekce dopravních značek většinou probíhá využitím konvolučních neuronových sítí. Cílem tohoto systému je upozornit řidiče, v případě přehlédnutí dopravního značení a následného nevědomého nedodržování dopravních předpisů. Technologie je v tomto případě již dobře zvládnutá a očekává se, že bude od roku 2022 povinná v každém novém vozidle prodaném v EU.



Obrázek 2-4: Ilustrace systému rozeznávání dopravních značek [10]

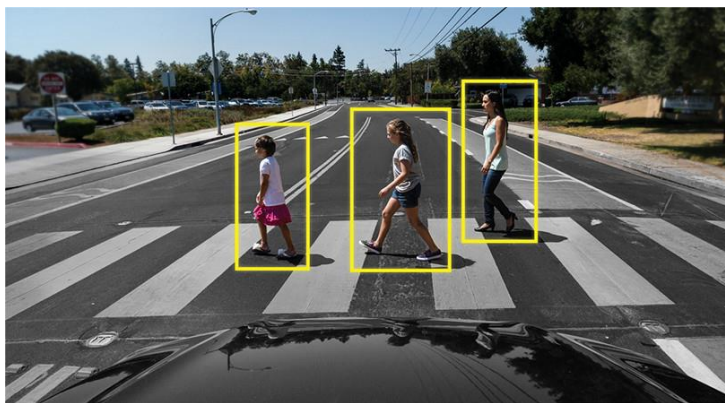
### **Systém zamezující kolizím:**

Systém zamezující kolizím byl na trh uveden v roce 2008. Jedná se o systém, který je schopen detekovat potenciální čelní srážku s jiným vozidlem. Podobně jako adaptivní tempomat, získává informace o okolí vozidla pomocí radaru nebo laseru. Jeho výhodou je kratší reakční čas oproti člověku. Systém je také schopen zabránit nehodě v situacích, kdy řidič nerozezná hrozící riziko nehody z důvodu nepozornosti, případně špatných světelných nebo povětrnostních podmínek. Tento systém má být zanedlouho povinný ve všech nových vozidlech.

### **Systém detekce chodců:**

Systém detekce chodců je jeden z nejnovějších ADAS systémů a poprvé byl představen v roce 2010. Využívá radar a poznatky počítačového vidění pro detekci osob v sekvenci kamerových snímků. Existuje několik způsobů detekce, buď se detekuje celé tělo chodce, nebo se využívá detekce po částech těla. V případě, že systém detekuje chodce na silnici a řidič nezareaguje zpomalením, resp. zastavením vozidla, systém automaticky zahájí brzdění.

V současnosti se tento typ systému rozšiřuje do stále více vozidel, nezávislé testy ovšem poukazují na to, že se na ně nelze spoléhat a jejich schopnost detekce chodců je u některých výrobců poměrně špatná.



Obrázek 2-5: Ukázka detekce chodců v běžném provozu [11]

## **2.2 Automatizace řízení**

Pokročilé asistenční systémy, jak bylo vysvětleno, využívají několik zdrojů dat současně. Díky tomu si automobil vytváří v menší nebo větší míře přehled o dění ve svém blízkém i dalekém okolí. Tento proces ve své podstatě simuluje vnímání okolí řidičem při jeho jízdě v automobilu [12]. Na základě tohoto předpokladu lze mluvit o automatizaci řízení.

Pod automatizací řízení rozumíme děj, při kterém dochází k částečné nebo úplné náhradě řidiče v procesu řízení vozidla. To, jak moc jsou ADAS systémy vozidla schopné automatizovat řízení, záleží od jejich množství a schopnosti spolu efektivně spolupracovat. Samozřejmě, co se týče množství, musí se jednat o systémy založené na různých principech.

V současnosti se ADAS systémy uplatňují ve vozidlech v stále větší míře. Z tohoto důvodu vznikl požadavek na zadefinování jednotlivých úrovní automatizace.



Obrázek 2-6: Vybrané ADAS systémy a jejich pole vnímání okolí vozidla [12]

## 2.3 Stupně automatizace řízení

Stupně autonomního řízení poskytují přehled o jednotlivých úrovních automatizace řízení a to od plně manuálního ovládání vozidla až po plně autonomní jízdu. Toto rozdělení bylo publikováno v roce 2014 společností automobilových inženýrů (SAE) a obsahuje celkově 6 stupňů [13] [14].

### 2.3.1 Level 0 – Bez automatizace

Tento level popisuje běžné auto bez jakýchkoliv automatizačních systémů. V tomto módu musí řidič provádět všechny úkony související s řízením, jako je ovládání volantu, přidávání, brzdění atd.

### 2.3.2 Level 1 – Asistence řidiče

Na této úrovni řidič pořád obsluhuje většinu funkcí vozidla, ovšem s menší pomocí ADAS systémů. Tuto pomoc může představovat, např. adaptivní tempomat, systém nouzového brzdění nebo třeba také parkovací asistent. V současnosti lze většinu nově vyrobených vozidel řadit právě do této kategorie autonomního řízení.

### **2.3.3 Level 2 – Částečná automatizace**

Částečná automatizace umožňuje řidičům přenechat některé funkce řízení počítačům. ADAS systémy se mohou podílet na udržování vozidla v jízdních pruzích a převzít řízení především v případě dálkových jízd na dálnici, např. využitím dat ze systému adaptivního tempomatu.

U tohoto levelu automatizace je nutné, aby byl řidič pořád připraven zasáhnout a převzít řízení, jinak řečeno je pořád zodpovědný za řízení a jeho bezpečnost.

### **2.3.4 Level 3 – Podmíněná automatizace**

U podmíněné automatizace, již mluvíme o vozidlu, které použitím snímačů, radarů (např. LIDAR) atd. monitoruje celé své vnější okolí. To umožňuje ADAS systémům plně převzít kontrolu nad automobilem. Tento level automatizace dovoluje řidičům nasměrovat svou pozornost na jiné aktivity, jako třeba použití mobilního telefonu. Tyto výhody jsou ale podmíněny rychlostí, kterou se vozidlo pohybuje a také povětrnostními podmínkami prostředí. Výrobce obvykle garantuje funkčnost plně automatizovaného řízení do rychlosti 60 km/h. Při vyšších rychlostech nebo špatném počasí je již potřeba, aby byl řidič připraven převzít řízení v případě nutnosti podobně, jako je tomu u levelu 2.

Tento level automatizace tedy nevyžaduje za každých okolností pozornost řidiče, ovšem pořád vyžaduje jeho přítomnost v případě potřeby. V současnosti se úroveň 3 vyskytuje u několika nejnovějších vozidel od prémiových výrobců.

### **2.3.5 Level 4 – Vysoká automatizace**

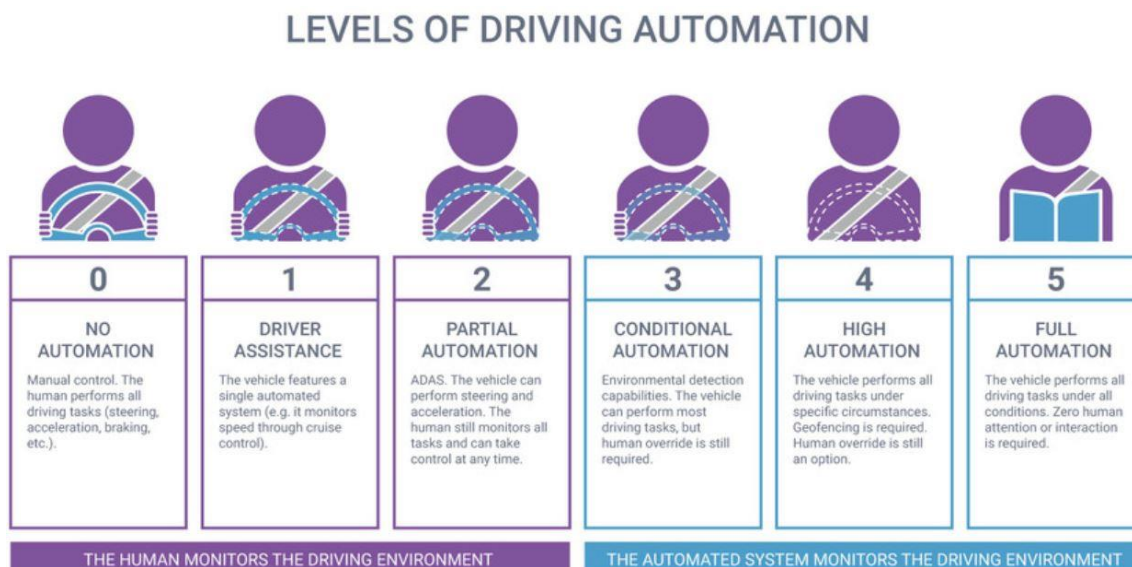
Na této úrovni automatizace je vozidlo schopno autonomní jízdy i ve vyšších rychlostech. Tato schopnost je však pořád ovlivněna vnějšími podmínkami a funguje jenom za dobrých podmínek. Level vysoké automatizace si nedokáže poradit s dynamickými dopravními podmínkami, jako jsou dopravní zácpy nebo zařazení se na dálnici.

Vozidlo s tímto typem automatizace je téměř úplně autonomní, pořád je zde ale potřeba zásahu řidiče ve specifických situacích. V praxi se s levelem 4 setkáváme zatím pouze u experimentálních vozidel.

### **2.3.6 Level 5 – Plná automatizace**

Vozidla s plnou automatizací řízení nevyžadují žádnou interakci s řidičem. Jsou schopny samostatného ovládání směru jízdy, akcelerace, brzdění a hlídání dopravních podmínek, včetně dopravních zácp. Neobsahují volant a řidič se mění na pasažéra.

Řidič má při tomto typu automatizace volnost bez nutnosti potřeby koncentrace na řízení. Tento level automatizace zatím nebyl dosažen, díky technologickému pokroku je ale pravděpodobné, že se ho svět časem dočká.



Obrázek 2-7: Grafický přehled jednotlivých úrovní automatizace řízení [14]

## 2.4 Pokročilé asistenční systémy v návaznosti na řidiče

Pokročilé asistenční systémy jsou doajista velkou pomocí pro řidiče při řízení vozidla a jsou schopny nahradit spoustu činností, které by jinak řidič musel vykonávat nebo situací, které by musel správně vyhodnotit. Na druhou stranu, mohou tyto systémy dávat řidiči falešný pocit bezpečnosti. Je faktem, že funkčnost a spolehlivost jednotlivých systémů není 100%-ní a nelze na ně úplně spoléhat. Tato hrozba se stává ještě aktuálnější, pokud se spojí několik navzájem spolupracujících systémů, které následně vytvoří určitý level autonomního řízení, jak bylo ukázáno v předešlé kapitole. Roli řidiče v takovém případě přebírají počítače a ten se tím pádem stává pouze pasažérem. V takové situaci je pro člověka snadné ztratit koncentraci a přestat sledovat okolní provoz. Řidič může mít tendenci dívat se do mobilu, začít si číst nebo dokonce usnout. Jak ale bylo zmíněno, současné ADAS systémy nejsou absolutně spolehlivé, jejich spolehlivost se může zhoršovat třeba za špatných povětrnostních podmínek. Rovněž platí, že tyto systémy nemusí být schopné reagovat na všechny nepředvídatelné situace.

Na základě těchto předpokladů je velmi důležité zabezpečit, aby byl řidič schopen a připraven v případě situací, se kterými si ADAS systémy neumí poradit, včas zasáhnout a převzít řízení. V opačném případě hrozí velké riziko nehod, kterým by šlo snadno zabránit. To, že je toto riziko realitou ukazuje i zvýšený počet zpráv, které popisují nehody způsobené právě spoléháním řidičů na různé formy autonomního řízení.

Pro eliminaci tohoto rizika vzniká v posledních letech nová kategorie ADAS systémů – systémy, které mají přímou návaznost na řidiče. Tyto systémy podobně, jako tradiční ADAS systémy využívají prostředky jako snímače, radary nebo kamery. Vstupní



data se ovšem nezískávají z okolí vozu, nýbrž z interiéru vozidla se zaměřením na samotného řidiče a jeho pohyby.

V dalších podkapitolách bude ukázán bližší pohled na jednotlivé existující typy ADAS systémů se zaměřením přímo na řidiče.

### **2.4.1 Systém rozeznávání opilého řidiče**

Alkohol za volantem je i v dnešní době velkým problémem. Reakce opilého řidiče na neočekávatelné situace jsou mnohem pomalejší. S rostoucí mírou opilosti se rovněž zhoršuje koordinace motorických schopností, zhoršuje se zrak a v neposlední řadě se otupuje schopnost myslet a vyhodnocovat potenciálně nebezpečné situace. Podle některých statistik je alkohol faktorem až v 30 % všech smrtelných dopravních nehod [15] a toto procento během let téměř vůbec neklesá.

Je tedy zřejmé, že i přes zákony namířené proti jízdě pod vlivem alkoholu se nelze vždy spoléhat na zdravý úsudek řidičů. Z toho důvodu vývojáři v posledních letech pracují na bezpečnostním systému, který je schopen opilost detekovat ještě předtím, než řidič nastartuje vozidlo. Existuje vícero principů detekce, jako nejperspektivnější se jeví tyto dva:

#### **Detekce alkoholu založená na dechu**

Detekce požití alkoholu z dechu řidiče je poměrně známá věc. Tento způsob vyžaduje vzorek výdechu řidiče, který je následně analyzován tzv. alkohol testerem. To ovšem vyžaduje kooperaci řidiče, což se u opilého řidiče sedajícího do vozidla nedá očekávat. Tento systém proto analyzuje vzduch, který řidič vydechne do prostředí interiéru. Ten se následně dostane dovnitř snímače, který změří koncentraci alkoholu a oxidu uhličitého přítomného ve vzorku. Běžně známý podíl oxidu uhličitého slouží jako indikátor množství koncentrace alkoholu ve vydechnutém vzduchu. Samotné měření využívá znalost toho, že jak molekuly alkoholu, tak oxid uhličitý absorbují infračervené světlo na určitých vlnových délkách [16].

Systém detekce alkoholu je prozatím pořád v prototypové fázi. Počítá se s umístěním snímače do prostředí přístrojové desky, případně na boční dveře řidiče.



Obrázek 2-8: Ilustrace analýzy alkoholu v dechu řidiče [16]

### Detekce alkoholu založená na doteku

Jiný způsob detekce alkoholu je založen na měření alkoholu pod povrchem kůže řidiče. Přesněji se jedná o obsah alkoholu v krvi detekovaný v tzv. krevních kapilárách. Měření probíhá nasvícením řidičovy kůže infračerveným zářením, které se následně dostane do jeho kůže. Část tohoto světla se odrazí a vrátí zpátky do snímače. V dalším kroku se na základě odražené vlnové délky určí chemická látka [17].

Tento systém je rovněž v prototypové fázi, snímač je možné umístit např. na startovací tlačítko motoru.



Obrázek 2-9: Ilustrace analýzy alkoholu v krvi řidiče [17]

### 2.4.2 Systém detekce ospalosti řidiče hlídáním jeho očí

Jiným bezpečnostním systémem, který je navázán přímo na řidiče, je systém detekující jeho ospalost, resp. únavu. Statistiky říkají, že až 20 % všech nehod souvisí s únavou řidičů při jízdě. Na některých typech cest se tato hodnota blíží k 50 %. Z těchto čísel je zřejmé, že včasná detekce ospalosti a varování řidiče může významně snížit riziko nehod.

Pro detekci ospalosti neboli únavy je potřebné definovat stavy, resp. veličiny, které mohou ospalost naznačovat. Ospalý řidič může např. zavírat oči, vykonávat

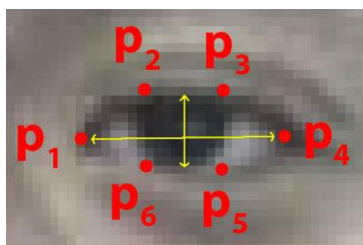


nekoordinované pohyby volantem atd. Na základě těchto předpokladů vzniklo několik systémů detekce ospalosti. V podkapitolách budou podrobněji rozebrány.

### Detekce ospalosti založená na hlídání očí řidiče:

Detekce založená na hlídání očí využívá pro vstupní data kameru zaměřenou na oči řidiče.

Princip spočívá v analýze mrkání, resp. zavírání očí řidičem. Ospalost je detekována, pokud řidič zavírá oči častěji, než stanovený práh, nebo pokud je zavírá na dobu delší, než je běžná doba mrkání. Tyto skutečnosti se určují na základě softwarového zpracování vstupních snímků, kde nejprve probíhá segmentace očí od zbytku obrazu. Po segmentaci je oblast oka vymezena 6 body, jak lze vidět na obrázku 2-10. Ze souřadnic těchto bodů lze spočítat tzv. poměr stran očí (eye aspect ratio).

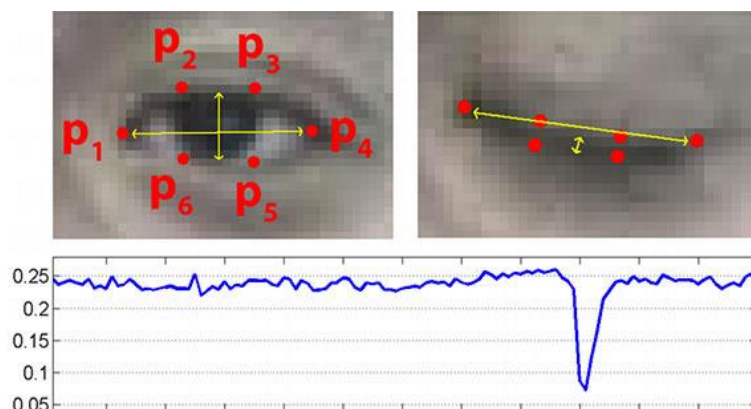


Obrázek 2-10: Ukázka detekce bodů vymežujících plochu oka [18]

Tento poměr se vypočte podle vzorce:

$$EAR = \frac{||p_2 - p_6|| + ||p_3 - p_5||}{2||p_1 - p_4||} \quad (1)$$

Při výpočtu hodnoty EAR je možné zjistit, že je tato hodnota u otevřeného oka přibližně konstantní. Pokud ovšem řidič oko zavře, hodnota EAR ihned klesá k nule. Pomocí těchto souvislostí lze tedy detekovat mrkání, resp. zavírání oka řidičem [18]. Princip detekce je ukázán na obrázku 2-11.



Obrázek 2-11: Ukázka změny hodnoty EAR při mrknutí oka [18]

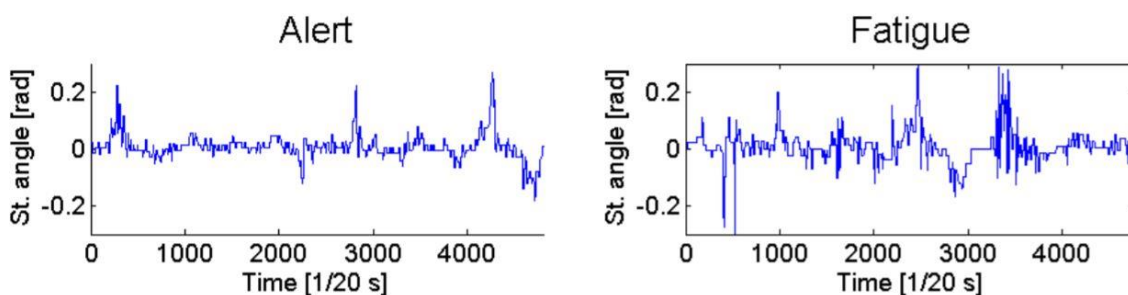
Detekce ospalostí analýzou očí je poměrně spolehlivá metoda, ovšem pouze za určitých podmínek. Jelikož se jedná o zpracování obrazových dat, musí být vstupní snímky dostatečně kvalitní. To znamená, že by v aute měl být dostatek světla i za tmy (nebo lze využít kameru s nočním viděním). Problémem může ale být i přílišné osvětlení řidiče slunkem. Situace se také komplikuje v případě, že řidič nosí brýle, v případě slunečních brýlí již tento systém detekce nelze použít vůbec.

### **Detekce ospalosti pomocí analýzy držení volantu:**

Jinou možností hlídání bdělosti řidiče je pomocí analýzy držení volantu a zatáčení. Vychází se z předpokladu, že unavený řidič reaguje na potřebu úpravy změny směru vozidla pomaleji, než řidič, který ospalý není. Zároveň když zareaguje, udělá nejspíše prudší korekci volantu z důvodu pomalejšího reakčního času a tudíž potřeby větší korekce. Efekty vyvolané únavou řidiče na jeho ovládání volantu by se dali shrnout takto: Méně malých, plynulých úprav volantu a více nepravidelných zig-zag pohybů volantu.

Pro měření pohybů volantu je možné využít dva různé principy. V prvním případě se data o pohybu volantu získávají ze snímače, který měří úhel natočení volantu nebo je možné využít kameru namířenou na volant. Ta ale se sebou přináší obvyklé problémy při snímání obrazu, spolehlivější cesta je tedy použít snímač úhlu natočení. Rovněž důležité je snímat úhel natočení s dostatečnou frekvencí. Platí, že reakce řidiče se nemůže měnit častěji než s frekvencí 50 Hz (každých 0,02 s), frekvence vzorkování by tedy měla být alespoň 50 Hz [19].

Rozdíl mezi aktivním řidičem a řidičem unaveným se při použití systému v automobilu projeví v grafu, jak ukazuje jedna ze studií, kde u unaveného řidiče dochází k dosahování větších hodnot korekce volantu, které se ale provádějí méně často [20].



Obrázek 2-12: Průběh korekcí volantu v čase - aktivní vs. ospalý řidič [20]

### **Detekce ospalosti založená na monitorování životních funkcí řidiče:**

Další a asi nejvíce kontroverzní metodou je detekce ospalosti založená na monitorování fyziologických funkcí člověka. Zde se jedná o tzv. elektrookulografii (EOG) a elektroencefalografii (EEG).

Elektrookulografie využívá dvou elektrod umístěných okolo očí. Oko se chová jako dipól s tím, že jeho rohovka představuje kladnou část dipólu, sítnice zápornou. Potenciál dipólu se mění při natočení očí, ale rovněž při mrknutí. Samotné mrknutí je pak v signálu možné vidět jako náhlou změnu amplitudy signálu, která trvá krátkou dobu. Z tohoto signálu je poté možné extrahovat parametry, jako jsou frekvence mrkání, amplituda vyjádřená v mV, která určuje míru zavření oka a také čas v ms. Ten určuje, jak dlouho zavření oka trvá [21]. Tyto parametry se pak vyhodnocují stejným způsobem, jako u video-detekce ospalosti řidiče.

Velkou výhodou elektrookulografie je invariantnost vůči vnějším světelným podmínkám. Nutnost nošení elektrod na obličeji řidiče [22] ji ale dělá nevhodnou pro komerční využití v automobilové dopravě a své využití nachází především v medicíně.



Obrázek 2-13: Ukázka použití elektrod pro provedení EOG [22]

Druhá metoda, která analyzuje fyziologické parametry člověka, je metoda elektroencefalografie (EEG). Jedná se o neinvazivní metodu vyšetření elektrické aktivity centrálního nervového systému. Snímání je provedeno opět pomocí elektrod, které jsou umístěny na povrchu hlavy a sledují signály z neuronů. Ospalost a zavřené oči v EEG charakterizuje zvýšené množství tzv.  $\alpha$  signálů o frekvenci 8 až 12 Hz a  $\Theta$  signálů o frekvenci 4 až 8 Hz. Zároveň dochází k poklesu množství  $\beta$  signálů (12-26 Hz), které jsou spojeny s aktivní koncentrací [23].

Problémem EEG může být slabá amplituda signálu, který jenom těžce prochází přes málo vodivou lebku a následný problém detekce hledaných frekvencí. Podobně jako u EOG, ani EEG není vhodná pro nasazení v automobilovém průmyslu.

### **Detekce ospalosti určená polohou vozidla vůči jízdním pruhům**

Jedná se o stejný systém, jako je systém varování před vybočením z jízdního pruhu, který byl popsán v kapitole 2.1. Jeho smysl jako detekce ospalosti se uplatňuje u vozidel, které nemají zároveň systém udržování vozidla v jízdním pruhu.

## 2.5 Navrhované systémy kontroly řidiče

V předešlých kapitolách byl uveden přehled i rozbor různých bezpečnostních systémů. Cílem diplomové práce je tuto množinu rozšířit navržením a vytvořením systémů pro sběr dat, které budou sloužit jako základ pro nové bezpečnostní systémy, hlídající pozornost řidiče způsoby, které dosud nebyly společně v komerční sféře implementovány. Tyto systémy se uplatní především z důvodu rozšíření vyšších stupňů automatizace ve vozidlech již v nejbližších rocích. Budou tedy navázány jak na řidiče, tak na dostupný stupeň automatizace. Bude se jednat o následující systémy:

### **Detekce ospalosti řidiče na základě mrkání a zívání**

Navrhovaný systém detekce ospalosti nebude hlídat pouze zavírání očí, nýbrž se bude také zaměřovat na detekci zívání. Pro tento způsob detekce se využije podobného principu jako u hlídání zavírání očí.

*Určeno pro level automatizace: 0 až 4*

### **Detekce připravenosti řidiče převzít řízení**

Systém bude určen pro hlídání pohybů řidičova těla. Upozorní řidiče na náklon do jedné ze stran, dopředu nebo dovrchu. Systém bude dále kontrolovat, zdali nenatáčí hlavu do stran a jestli netelefonuje bez použití technologie hands-free. Řidič bude ale mít povoleno zvednout jednu z rukou, pokud se bude dívat na vozovku.

*Určeno pro level automatizace: 0 až 4*

### **Detekce držení volantu řidičem**

Systém bude určen pro situace, které vyžadují, aby měl řidič ruce pořád na volantu, a upozorní řidiče v případě, že nedrží volant po delší dobu. Systém lze považovat za alternativu ke kapacitnímu principu detekce držení volantu.

*Určeno pro level automatizace: 0 až 2*

### **Kontrola umístění noh v blízkosti pedálů**

Navrhovaný systém bude sledovat, jestli má řidič své nohy pořád v prostoru plynového a brzdového pedálu. V případě, že tomu tak není, vyšle upozornění.

*Určeno pro level automatizace: 0 až 4*

### 3 MĚŘICÍ SESTAVA

Tato kapitola se věnuje návrhu měřicí sestavy. Konkrétně je zde nejprve provedena analýza jednotlivých typů kamer pro snímání vstupních snímků, na jejíž základě je zvolen přesný typ kamery. Poté se kapitola věnuje návrhu jednotlivých měřicích scén, které je potřebné zadefinovat v kontextu tvorby systémů. Na závěr je možné dočíst se o praktických problémech a procesu snímání vstupních snímků vybranou kamerou.

Aby byl proces vytváření navrhovaných algoritmů co nejjednodušší a algoritmy co nejspolehlivější, je potřeba zamyslet se nad samotnou měřicí sestavou. Ta bude zahrnovat jak měřicí scény, tak kameru, kterou budou měřicí scény snímány.

#### 3.1 Volba typu kamery a možné problémy

Pro výběr vhodného typu kamery je dobré popřemýšlet, v jakých podmínkách bude snímání probíhat a také jaká omezení jednotlivé typy kamer přináší. Nejběžnějším typem kamery je kamera snímající dvourozměrný obraz, určená do denního světla. Je ovšem potřeba brát v potaz fakt, že řidiči vozidel zcela běžně řídí ve večerních nebo nočních hodinách. V takových případech je zřejmé, že běžná kamera nebude postačovat, protože na snímcích nebude možné vidět požadované detaily. Problém je možné řešit dvěma způsoby.

První variantou je měřicí scénu osvětlovat umělým zdrojem viditelného světla. Tento typ řešení nejspíše půjde použít pro systém kontroly přítomnosti nohou. Scéna, kterou bude potřeba pro tento systém sestavit, bude dozajista obsahovat prostor pedálů a pravdou je, že někteří výrobci tento prostor již osvětlují, i když z dekorativních důvodů. Umělé osvětlení lze teoreticky také použít pro osvětlování prostoru volantů. Nelze ho však využít pro systémy, které budou pracovat se snímky samotného řidiče. Světlo by mu totiž svítilo přímo do očí a tím ho oslepovalo.

Pro řešení tohoto problému je lepší použít jiný typ kamery. Nabízí se kamery s tzv. nočním viděním. Funkce nočního vidění vysílá pro osvětlení měřicí scény světlo v infračervené oblasti. Scéna se tím pádem pro kameru již nejeví jako tmavá, ovšem lidské oko není schopné toto světlo vidět, tudíž se tím řeší problém oslepení. Výsledkem snímání obrazu v režimu nočního vidění je monochromatický obraz, ten ale nebrání detekci hledaných částí těla řidiče. Příklad porovnání snímání bez a s nočním viděním se nachází na obrázku 3-1.



Obrázek 3-1: Snímání bez a s nočním viděním [24]

Zcela jiným problémem, který může nastat, se týká samotné koncepce kamery. Běžné kamery, jak bylo zmíněno, poskytují dvourozměrný obraz. Hledané objekty, jako je např. hlava řidiče, jeho ruce nebo nohy, nemusí být možné segmentovat v případě, že kontrast mezi hledanými oblastmi a pozadím nebude dostatečně velký. Je tedy velký předpoklad, že takové řešení nebude dostatečně spolehlivé.

Možným řešením je využití principu stereovidění, resp. pasivní triangulace. Stereovidění využívá dvou kamer umístěných vedle sebe v určité vzdálenosti, které zachytávají zvolený objekt. Na základě platných vztahů, které popisují pasivní triangulaci lze určit třetí souřadnici jednotlivých bodů popisujících hledaný objekt. Výstupem takových kamer je obraz ve 3D. Díky tomu je možné zcela spolehlivě oddělit hledané oblasti od nepotřebného pozadí. Příkladem stereo kamery je kamera Bumblebee 2 od výrobce Flir nebo kamera RealSense od společnosti Intel.

Alternativou k stereovidění je použití nástroje založeném na principu měření doby letu paprsku (TOF). Příkladem je zařízení kinect od společnosti Microsoft. Tato technologie využívá infračervený laser. Na základě jeho zpracování provádí výpočty hloubkové mapy analyzovaného objektu. Technologie ovšem funguje především v interiérových aplikacích. V exteriéru jsou spolehlivější stereo kamery. Měřicí scény této práce se ale vyskytují pouze v interieru, je tedy možné použití obou technologií.



Obrázek 3-2: Stereo kamera Bumblebee 2 a Microsoft Kinect

Po zvážení všech faktorů lze říct, že typ kamery, který by byl nejlepší pro navrhované systémy, je stereo kamera obsahující noční vidění. Pokud by stereo kamera

neobsahovala noční vidění, lze vyslovit předpoklad, že navrhované segmentační metody budou fungovat i pro snímky, které by byly vytvořeny s pomocí funkce nočního vidění.

## 3.2 Stereo kamera Intel RealSense

Pro účely této práce byla zvolena a zakoupena stereo kamera Intel RealSense D415. Jedná se o kameru se snímacím rozsahem od 0,3 m, nejlépe ovšem snímá v rozsahu od 0,5 m do 3 m. Tento rozsah je pro navrhované měřicí scény dostatečný. Hloubkový modul kamery je díky principu aktivních infračervených vln schopen pracovat za každých světelných podmínek. Co se týče rozměrů, kamera má délku 10 cm, šířku 2 cm a výšku 2,3 cm. Její rozsah, noční vidění a malé rozměry ji tedy činí vhodnou pro snímání potřebných měřicích scén v prostoru vozidla.



Obrázek 3-3: Stereo kamera Intel RealSense [25]

Hloubkový modul kamery je schopen snímat až 90 snímků za sekundu (fps), u RGB modulu se jedná o hodnotu 30 fps. Pro možnou synchronizaci hloubkového modulu se standardním RGB modulem bude tedy nutné snímat s rychlostí 30 fps.

Také rozlišení se u jednotlivých modulů liší, u RGB modulu je možné snímat ve full-HD (1920x1080), u hloubkového modulu se jedná o hodnotu odpovídající HD (1280x720). Obě rozlišení jsou dostatečně kvalitní, spíše se lze domnívat, že pro účely segmentace bude postačovat i rozlišení mnohem menší, třeba SD. Výhodou pak bude menší výpočetní náročnost pro navržené algoritmy a zvýšení jejich rychlosti provedení, což je při snímání videa velmi důležitý atribut.

V tabulce níže je uveden kompletní přehled nejdůležitějších parametrů zvolené kamery [25].

Vlastnosti	Prostředí	Ideální rozsah snímání
	Interiér/Exteriér	0,5 – 3 m
	Rozměry	Konektory
	99 mm x 20 mm x 23 mm	USB-C 3.1 Gen 1

	<b>Technologie snímání</b>	<b>Procesor</b>
	Rolling Shutter	RealSense Vision Procesor D4
<b>Hloubkový modul</b>	<b>Technologie snímání</b>	<b>Zorné pole (FOV)</b>
	Active IR stereo	64° x 41° x 72° ±3°
	<b>Minimální vzdálenost</b>	<b>Rozlišení</b>
	~0,45 m	max. 1280x720
	<b>Přesnost</b>	<b>Počet snímků za sekundu</b>
	<2 % na 2 m	max. 90 fps
<b>RGB modul</b>	<b>Rozlišení</b>	<b>Zorné pole (FOV)</b>
	1920x1080	64° x 41° x 77° ±3°
	<b>Rozlišení RGB snímače</b>	<b>Počet snímků za sekundu</b>
	2 MPx	30 fps

Další předností, kterou tato stereo kamera nabízí je možnost práce s tzv. Intel Realsense SDK (software development kit). Jedná se o soubor softwarových nástrojů od společnosti Intel, které zabezpečují podporu pro různé operační systémy, rozhraní i nejběžnější programovací jazyky, včetně programovacího jazyka Python, který bude tato práce využívat.

### 3.3 Návrh měřicích scén

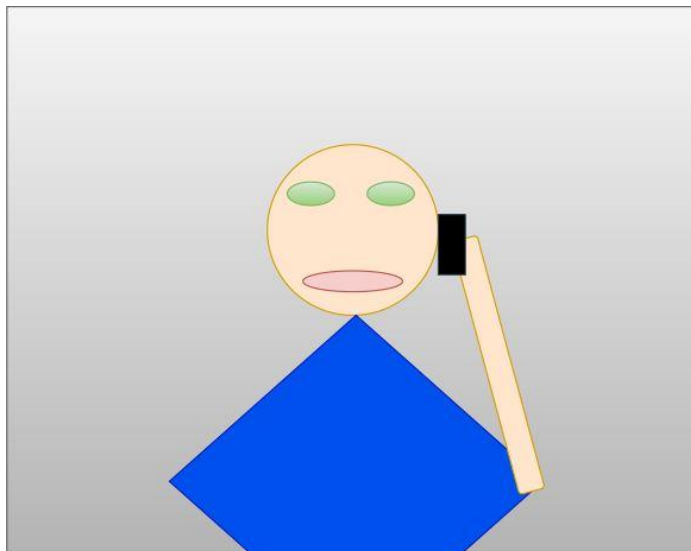
Po výběru kamery je dále potřeba promyslet vhodné uspořádání a počet měřicích scén. Podle načrtnutých systémů z podkapitoly 2.5 je možné usoudit, že scén, které je potřeba nasnímat je více. Cílem zároveň je, aby bylo kamer pro snímání co nejméně. Na základě úvah lze měřicí sestavu rozdělit na tři scény.

#### 3.3.1 Měřicí scéna č. 1 – Pohled na obličej a vrchní část těla řidiče

Tuto scénu bude snímat kamera umístěna na palubní desce tak, aby dokázala nasnímat obličej a vrchní část těla řidiče. Také platí, že zorné pole kamery musí pokrýt celou vrchní část řidiče při různé vzdálenosti sedadla od kamery. Musí být zároveň umožněno snímání okolí řidiče do šířky alespoň 0,5 m z každé strany, a to z důvodu detekce náklonu do stran.

Důležitou oblastí v rámci této měřicí scény bude především obličej, na základě kterého bude možné sestavit systém pro detekci únavy řidiče (hlídání očí a úst). Dále se tato scéna využije pro systém detekci připravenosti řidiče převzít řízení hlídáním natočení jeho hlavy, resp. celého těla. Model této měřicí scény může vypadat přibližně jako na obrázku 3-4.

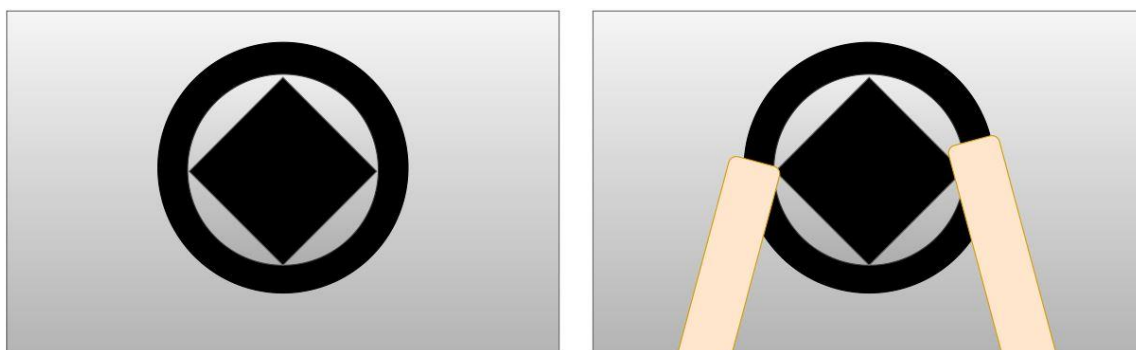




Obrázek 3-4: Návrh měřicí scény typu pohled na obličej řidiče

### 3.3.2 Měřicí scéna č. 2 – Pohled na volant řidiče

Tato měřicí scéna bude zobrazovat pohled na volant vozidla. Bude sloužit jako zdroj snímků pro systém zaměřený na hlídání držení volantu v případě, že si to level automatizace vyžaduje. Modely, které popisují tento typ scény, lze vidět na obrázku 3-5. Kamera bude nejspíše umístěna na stropu vozidla nad hlavou řidiče.



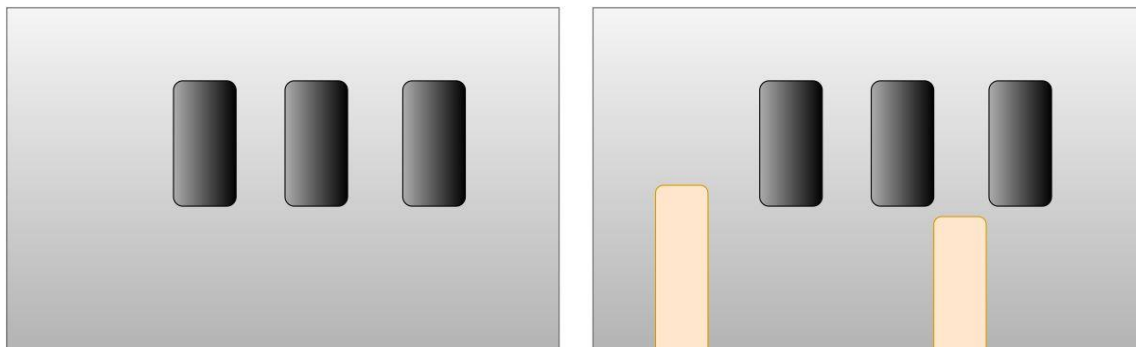
Obrázek 3-5: Návrh měřicí scény typu pohled na volant

Tuto scénu by teoreticky bylo možné snímat i z pohledu kolmo na volant, kde by šlo zároveň hlídat náklony hlavy, případně za jistých okolností i nohy řidiče. Při použití stereo kamery by ale hrozilo, že se kamera dostane mimo měřicí rozsah (hlava příliš blízko ke kameře), navíc tento pohled neposkytuje informaci o obličeji řidiče, nelze tím pádem tyto dvě scény spojit.

### 3.3.3 Měřicí scéna č. 3 – Pohled do prostoru pedálů

Poslední měřicí scéna, kterou bude tato práce využívat, bude měřicí scéna, která bude obsahovat pohled do prostoru pedálů, resp. na nohy řidiče. Scéna je potřebná pro systém, který hlídá řidičovu připravenost ovládat jednotlivé pedály. Kamera se bude

nejspíše nacházet pod spodní částí okolí volantu. Takové umístění by snímalo scénu podobnou, jako je ukázáno na modelu na obrázku 3-6. Vlevo lze vidět situaci, kdy řidič není připraven stlačit pedály, vpravo jsou již jeho nohy připravené.



Obrázek 3-6: Návrh měřicí scény typu pohled do prostoru pedálů

I zde se nabízí otázka jiných umístění kamery. Kamera by také mohla snímát pohled z boku, v takovém případě by ale docházelo k překrývání nohou. Při umístění kamery do spodní části sedadla by nohy řidiče mohly zakrývat objektiv kamery. Nakonec, při postavení kamery do prostoru za pedály nebo pod pedály by se narazilo na problém s měřicím rozsahem.

### 3.4 Nasnímání testovacích snímků

Po provedení návrhu měřicích scén je v dalším kroku možné přistoupit k nasnímání snímků, které budou sloužit k testování navrhovaných systémů. Zde se jako největší problém jeví vhodné umístění kamery/kamer.

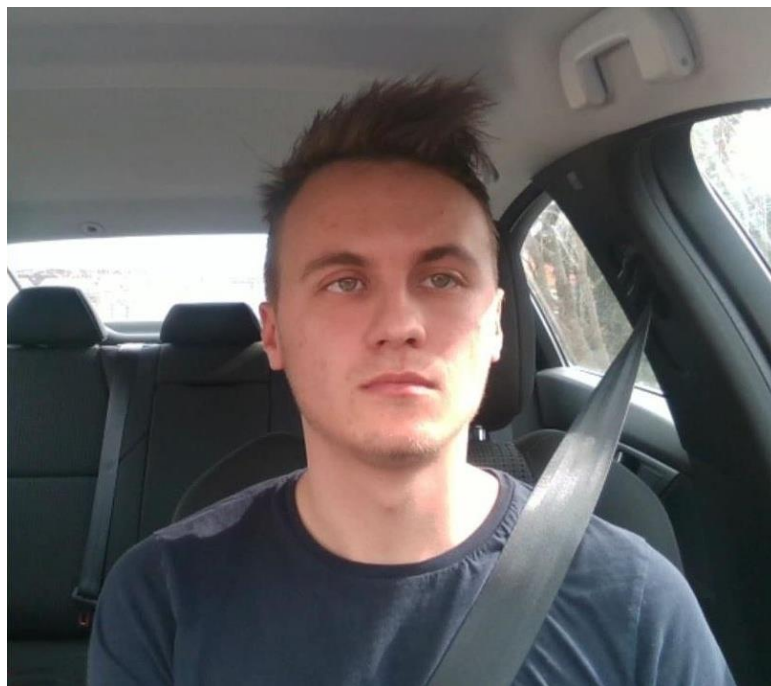
Při vývoji nového modelu vozidla v automobilovém průmyslu by se všechny tři potřebné kamery nainstalovaly do konstrukce interiéru tak, aby byly co nejméně nápadné a zároveň, aby snímaly potřebné scény co nejlépe. V praxi si to pro měřicí scénu č. 1 lze představit jako umístění kamery na horní okraj palubní desky za prostorem volantu, nebo jako součást horizontálního sloupku, který drží horní okraj čelního skla. U scény č. 2, ve které se snímá okolí volantu, by se jednalo o umístění kamery na strop vozidla, zhruba v místě, které určuje poloha řidičovy hlavy. Kamera by zároveň musela být natočena pod úhlem, který by dostatečně obsáhnul prostor volantu. U scény č. 3 je nejpravděpodobnější variant, ve kterém by se kamera umístila buď do konstrukce sedadla, nebo ke konstrukci volantu ze spodní strany.

V rámci této práce je k dispozici několik aut, pochopitelně však není možné zasahovat přímo do jejich konstrukce interiéru. Proto je potřeba najít kompromisní řešení, která se budou co nejvíce přibližovat popisovaným umístěním kamer a tím pádem splňovat požadavky na snímání požadovaných scén.

V zásadě lze uvažovat nad dvěma možnostmi, buď je možné kameru na danou pozici přilepit, např. lepicí páskou, nebo se kamera postaví na stativ. Po vyzkoušení obou variant se ovšem jasně ukázalo, že mnohem lepší řešení je používat stativ. Při pokusu o přilepení kamery na strop vozidla (měřicí scéna č. 2), nebo o přilepení kamery ke spodní části volantu (měřicí scéna č. 3) bylo téměř nemožné rovinu kamery nastavit tak, aby šlo přesně snímat požadované oblasti a zároveň, aby kamera držela na daném místě. Při použití stativu není obecně problém nastavit a udržet kameru ve zvolené poloze. Obtíž je v nalezení způsobu, jak stativ pro jednotlivé scény postavit tak, aby byl kameře umožněn výhled na danou scénu. Pro měřicí scénu č. 1 (pohled na řidiče) byl využit malý stativ, který se spolu s kamerou položil na palubní desku vozidla, viz obrázek 3-7. Toto řešení umožňuje snímat oblast řidiče a jeho okolí tak, jak to požaduje návrh. Výslednou podobu měřicí scény č. 1 je možné vidět na obrázku 3-8.

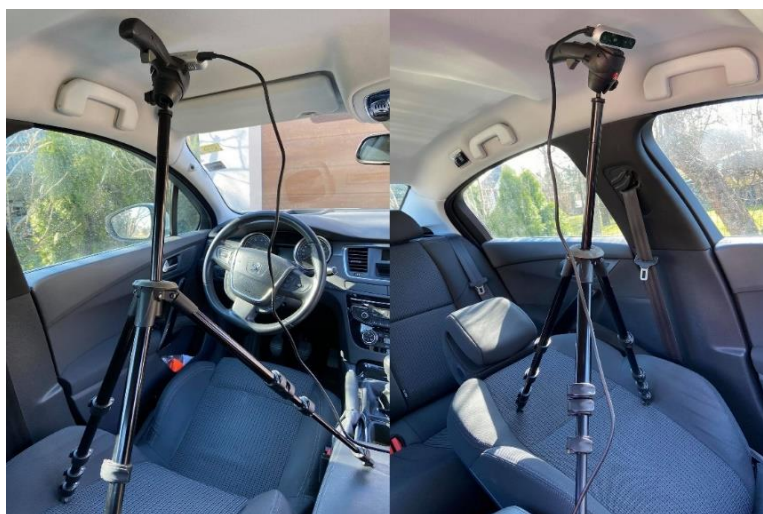


Obrázek 3-7: Poloha kamery na malém stativu pro snímání měřicí scény č. 1



Obrázek 3-8: Měřicí scéna č. 1

U měřicí scény, ve které je potřebné snímat volant, byla situace složitější. Stativ s kamerou by totiž ideálně musel být postaven na sedadlo řidiče. To je ale nereálné, jelikož se na sedadle nachází řidič, u kterého je v rámci této měřicí scény potřeba snímat, zdali má ruce na volantu nebo nikoliv. Jako prvotní řešení bylo zvoleno postavit stativ na zadní sedadlo za řidičem a kameru naklonit tak, aby byla nasměřovaná na snímání oblasti volantu. Toto řešení selhávalo v případě řidiče s vyšším vzrůstem, kdy docházelo k překrytí zorného pole kamery jeho hlavou. Nakonec se nejvíce osvědčilo sklopit vrchní část sedadla, jak ukazuje obrázek 3-9 a pak tuto sklopenou část použít jako platformu pro postavení stativu. Poté šlo nasnímat snímky pro měřicí scénu č. 2, viz obrázek 3-10.



Obrázek 3-9: Poloha kamery na velkém stativu pro snímání měřicí scény č. 2



Obrázek 3-10: Měřicí scéna č. 2

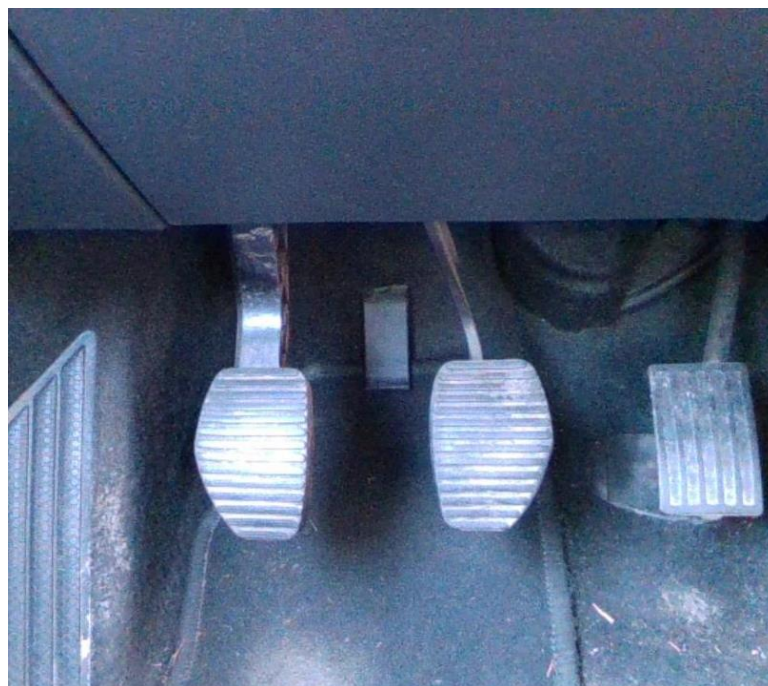


Samozřejmě toto řešení by z bezpečnostních důvodů nebylo možné aplikovat při reálném používání vozidla. Proto lze tento způsob snímání použít pouze pro experimentální účely testování algoritmů, které budou navrženy v příští kapitole.

U měřicí scény č. 3 (snímání nohou) se po otestování nejvíce osvědčilo postavit stativ mezi nohy řidiče a kameru naklonit pod takovým úhlem, aby snímala právě oblast pedálů a jejich okolí (obrázek 3-11). I zde platí, že tento princip snímání nelze v reálném provozu aplikovat.



Obrázek 3-11: Poloha kamery na velkém stativu pro snímání měřicí scény č. 3



Obrázek 3-12: Měřicí scéna č. 3

## 4 NÁVRH A REALIZACE METOD DETEKCE ŘIDIČE A JEHO POHYBŮ

V této sekci bude popsán výběr a realizace metod vhodných pro řešení problémů segmentace a detekce hledaných částí řidiče.

Při zamyšlení se nad principy navrhovaných systémů pro sběr dat, je zřejmé, že podstata práce spočívá ve spolehlivé detekci a segmentaci řidiče nebo zvolených částí jeho těla. Následně je potřeba určit odchylky od požadované polohy nebo úhlu natočení.

Jak bylo naznačeno v úvodních kapitolách, práce jako zdroj dat využívá kamerové snímky, ze kterých následně získává informace pro navrhované systémy. V rámci kapitoly 3 *Měřicí sestava* byl zároveň proveden rozbor možných typů kamer, které by šlo použít pro snímání. I přes jasné výhody stereo kamer je v rámci této práce vyzkoušeno i několik segmentačních metod pro běžné dvourozměrné snímky. Je to z toho důvodu, že stereo kamera nebyla v rámci zpracovávání diplomové práce dostupná hned od začátku. Zároveň vytvoření těchto prvotních metod slouží k hlubšímu pochopení problémů se segmentací a poukazuje na předpokládané problémy spojené se segmentací u dvourozměrných vstupních snímků.

### 4.1 Základní rozbor

Na základě kapitoly č. 3 je možné utvořit si představu o tom, které části obrazu bude nutné detekovat a sledovat pro implementaci jednotlivých systémů. Jelikož byly definovány tři měřicí scény, je potřeba se v kontextu vytvářených systémů nad každou zamyslet zvlášť.

První měřicí scéna slouží pro snímky, které bude používat jak **systém pro detekci ospalosti**, tak **systém pro detekci připravenosti řidiče převzít řízení**.

Pro detekci ospalosti bude postačovat správně detekovat oblast očí a úst. Následně se může použít princip detekce mrkání, který byl popsán v kapitole č. 2. Podobný princip bude možné využít i pro detekci zívání.

Pro realizaci systému detekce připravenosti řidiče převzít řízení, se na první pohled nabízí možnost segmentace celé viditelné části člověka. Poté je možné zaměřit se na měření odchylek náklonu řidiče do jednotlivých stran. Aby bylo s čím odchylky srovnávat, bude nejspíše nutné stanovit si při začátku měření referenční snímek, tj. pozici řidiče, ve které se dívá ve vzpřímené poloze na vozovku. Pozici by šlo odhadnout na základě známé polohy kamery a sedadla, výšku řidiče ale bude nutné změřit pro každého řidiče zvlášť. Při dalším zamyšlení se je možné říct, že nejjednodušším řešením by mohlo být zaměřit se pouze na obálku hlavy. Důvod je ten, že pokud řidič provede

jakýkoliv pohyb trupem, zákonitě to bude znamenat, že se i jeho hlava odkloní od normálu. Dalším cílem tohoto systému má být zamezení pohledů řidiče do stran. U tohoto problému by šlo zaměřit se na změny v šířce obálky hlavy, resp. na změnu v parametrech, které popisují oblast očí a úst. Pro kontrolu, jestli řidič netelefonuje, je rovněž potřebné zaměřit se na oblast hlavy a jako možné řešení detekce by mohlo být měření šířky vysegmentované části hlavy nebo opět měření šířky plochy, kterou vysegmentovaná část zabírá oproti pozadí. Segmentovat hlavu v prostoru vozidla je u dvourozměrných snímků v zásadě možné, a to díky specifické barvě kůže člověka, která obvykle nesplývá s pozadím v automobilu. Přesto lze čekat velkou citlivost na světelné podmínky. Grafickou reprezentaci měřicí scény č. 1 ve formě cílových oblastí jednotlivých subsystémů, které ve výsledku tvoří dva různé systémy, lze vidět na obrázku 4-1.



Obrázek 4-1: Analýza měřicí scény č. 1 v kontextu navrhovaných systémů

Zbylé systémy, konkrétně systémy pro **detekci držení volantu** a **detekci nohou** využívají měřicí scény č. 2, resp. č. 3. I když se jedná o rozličné scény, mohly by tyto systémy být společně realizovány na podobném principu. V praxi to znamená, že je potřeba segmentovat viditelné části rukou, resp. nohou od zbytku zaznamenávané plochy a následně měřit odchylky ve velikosti vysegmentovaných ploch oproti referenčním snímkům. Pod referenčními snímky si je možné představit snímky, kde se ruce, resp. nohy nenachází.

Při práci s dvourozměrnými snímky se může segmentace těchto částí ukázat jako neřešitelná. Pokud nebude mít řidič dostatečně kontrastní oblečení, resp. obuv, když se jedná o detekci nohou, segmentace nejspíše selže. Již předem lze tedy usoudit, že tento typ systému nejspíše nepůjde vytvořit spolehlivě bez použití stereo kamery.

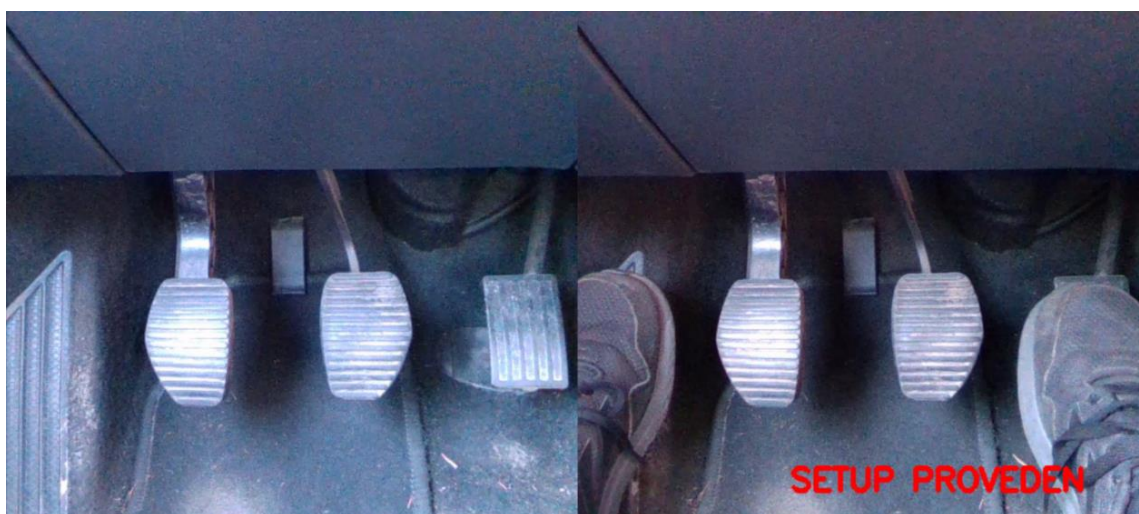


Grafická analýza měřicí scény č. 2 se nachází na obrázku 4-2. Vlevo se nachází tzv. referenční snímek, který bude potřeba vytvořit při inicializaci systému, vpravo je vyobrazena situace, kdy řidič volant drží v obou rukou. Ta může nastat právě až poté, co se provede nasnímání referenčního snímku – provede se setup. Následně, pokud se tedy povede oddělit pozadí (oblast interieru vozidla od popředí - ruce a případně volant), mělo by být možné, jak bylo zmíněno, na základě celkových vysegmentovaných ploch, tyto dvě situace spolehlivě rozlišit.



Obrázek 4-2: Analýza měřicí scény č. 2 – systém detekce držení volantu

Obdobně lze provést grafickou analýzu i pro měřicí scénu č. 3. V referenčním snímku pro detekci nohou se bude nacházet oblast pedálů bez řidičových nohou. Jakmile se řidičovy nohy ve snímku objeví (a bude možná segmentace od pozadí spolu s pedály), bude i zde možné rozlišovat situace, kdy je řidič připraven stlačit pedály a kdy není.



Obrázek 4-3: Analýza měřicí scény č. 3 – systém detekce připravenosti stlačit pedály



## 4.2 Prvotní metody pro vstupní snímky z běžné kamery

Jelikož nebyla stereo kamera dostupná od počátku psaní této práce, prvotní metody pro řešení navrhovaných algoritmů využívají snímky měřicích scén z běžné kamery, která snímá dvourozměrně. Ty mohou vypadat mírně odlišně od datasetu, který je popsán v kapitole *Měřicí scéna*. Pro takové snímky je možné implementovat několik různých typů segmentace.

Jednotlivé metody segmentace obecně pracují na principu klasifikace nebo seskupení obrazu do několika nesouvislých částí seskupením pixelů tak, aby tvořily oblast homogenity na základě daných charakteristik jednotlivých pixelů. Může se jednat o parametry jako jasová úroveň, barva, textura atd. Novější přístup zahrnuje „učení se“ vzorů z referenční databáze vstupních obrázků a následné hledání naučených vzorů v analyzovaném obrázku. Každá z metod má své výhody i nevýhody. Je proto těžké dopředu určit, která z metod se bude hodit nejvíce pro segmentaci řidiče, resp. jeho částí těla.

Segmentační metody pro dvourozměrné snímky lze na základě principů rozdělit do dvou hlavních skupin. První skupina představuje „old-school“ metody, které se v minulosti běžně používaly, nejsou ovšem tak efektivní jako metody strojového učení, ale jsou mnohem méně náročné na výpočetní výkon. Patří zde tyto metody [26]:

- **Prahování:** Rozdělí obrázek na popředí a pozadí. Oddělení pixelů probíhá na základě zadané prahové hodnoty. Výsledkem je binární obraz. Prahování může probíhat buď s jedním prahem globálně pro celý obraz, nebo se zvoleným prahem pro malé oblasti obrazu – adaptivní prahování. Prah lze zvolit, např. jako průměr nebo medián pixelů prahované oblasti.
- **Shlukování:** Tento algoritmus iterativně rozděluje pixely do jednotlivých shluků na základě délky k centroidům shluků.
- **Metody založené na histogramu:** Používají histogram obrázku a pracují s předpokladem, že pixely reprezentující pozadí patří do jiného bloku hodnot v histogramu než pixely popředí.
- **Použití hranových operátorů:** Identifikuje velké změny jasové funkce použitím operace konvoluce hranového operátoru se vstupním obrázkem.

Do druhé skupiny segmentačních metod lze zařadit metody založené na strojovém učení. Tyto metody se rozšířily především v posledních letech z důvodu značného zvětšení množství výpočetního výkonu v počítačích. Jak bylo zmíněno, tyto metody jsou schopny, na základě rozsáhlých datasetů, naučit se vzory, které jsou typické pro hledaný/segmentovaný objekt.

Toto rozdělení metod segmentace se uplatňuje při práci se standardními dvourozměrnými snímky (u 3D snímků se využívá jiný princip). V dalších podkapitolách bude proveden rozbor a realizace některých možných metod pro řešení jednotlivých systémů pro sběr dat.

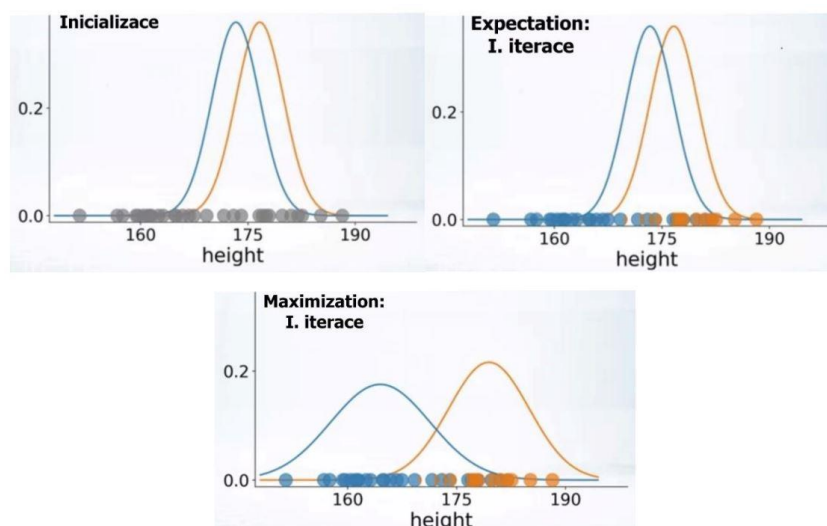
#### 4.2.1 Detekce metodou Gaussian mixture model

Metoda Gaussian mixture model neboli Gaussovský smíšený model je shlukovací metoda podobná metodě k-means. Shlukování si lze představit jako hledání oblastí obrazových bodů, které sdílí nějaké společné charakteristiky. Zatímco u metody k-means dochází k přiřazení každého bodu k právě jednomu shluku a označujeme ji za tvrdou shlukovací metodu, u metody Gaussian mixture model se jedná o tzv. měkkou shlukovací metodu. To znamená, že každý bod je přiřazen k jednotlivým shlukům s určitou pravděpodobností.

Pomocí této metody je možné oddělit jednotlivé homogenní vrstvy v obraze. Pro případ této práce by bylo vhodné oddělit oblasti řidičova těla, teda hlavu nebo oblast rukou na základě předpokladu, že obrazové body, které tvoří kůži řidiče, tvoří homogenní oblast, jež je zároveň odlišná od jiných oblastí v obraze. Nejprve je ale potřeba ozřejmit samotný princip metody.

#### Princip metody – algoritmus EM

Jak bylo zmíněno, u metody GMM je každý bod přiřazen k daným shlukům vždy s určitou pravděpodobností. Tato pravděpodobnost a samotné shluky jsou reprezentovány Gaussovými funkcemi. Gaussova funkce je definovaná svým středem, šířkou a výškou. Tyto parametry se iterativně mění zprůměrováním pravděpodobností jednotlivých bodů a jejich následným vážením (znovu podle pravděpodobnosti, že se daný bod nachází v daném shluku). V každém kroku se tedy mění hodnoty pravděpodobností, a to na základě polohy daného bodu vůči Gaussově funkci daného shluku. Tento algoritmus je známý jako Expectation Maximization (EM) Algorithm. Nejprve se tedy podíváme (**Expectation**), zdali daný bod vypadá, že by mohl patřit k danému shluku (vyjádření pravděpodobnosti). Následně aktualizujeme parametry příslušné Gaussovy funkce (**Maximization**) tak, aby odpovídala bodům, které jsou jí přiřazeny – patří k danému shluku. Platí, že body nejbližší ke středu Gaussovy funkce mají největší pravděpodobnost toho, že patří do daného shluku. U bodů vzdalujících se od středu pravděpodobnost klesá, na druhou stranu jim stoupá pravděpodobnost, že patří do jiného shluku [27]. Dobrou představu o principu GMM algoritmu nám poskytne obrázek č. 4-4.



Obrázek 4-4: Ukázka jedné iterace GMM algoritmu na datech výšky skupiny lidí

Zde můžeme vidět příklad hledání dvou shluků (dvě Gaussovy funkce) ve vstupních datech, které představují výšku skupiny lidí.

V případě detekce hledaných částí těla řidiče v kamerových snímcích je vhodné zamyslet se, kolik shluků – Gaussových funkcí je potřeba detekovat. Pokud by se využilo pouze dvou Gaussových funkcí, znamenalo by to, že se výstupní obrázek „rozdělí“ na dva segmenty. V ideálním případě, by se jednalo o oblast segmentovaného objektu a oblast pozadí. U vstupních snímků z navrhovaných systémů tomu tak ale nejspíše nebude, jelikož se zde homogenní pozadí spíše nevyskytuje. Alternativou může také být popsání obrázku nejprve bez řidiče a následné hledání odchylek.

## Testování metody GMM

Hned na začátku je nutno si uvědomit, že tato metoda není vhodná pro segmentaci očí nebo úst řidiče, jelikož se jedná o příliš malé oblasti, které navíc nejsou homogenní. Metoda se mnohem lépe uplatní při detekci hlavy, rukou nebo nohou řidiče.

### Detekce obličeje

Pro účel detekce obličeje bylo v prvním kroku potřeba určit, do kolika vrstev bude nejvýhodnější obrázek rozdělit. Tuto situaci lze pozorovat na obrázku 4-5, kde každý obrázek představuje výstup algoritmu GMM s jiným přednastaveným počtem použitých Gaussových funkcí. Lze říct, že už při použití dvou Gaussových funkcí je jasně vidět oddělenou oblast hlavy od pozadí, část pozadí ale splývá s hlavou. Podobná situace nastává při použití tří Gaussových funkcí. Teprve při použití čtyř, resp. pěti Gaussových funkcí je obraz rozdělen do shluků tak, že oblast popisující hlavu je jasně oddělena od oblastí pozadí.



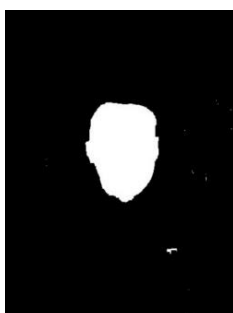
Obrázek 4-5: Rozdělení shluků za použití různého množství Gaussových funkcí

Situace nemusí být vždycky tak příznivá. Pokud bude mít řidič oblečení, které bude barvou příliš podobné barvě jeho kůže nebo bude pozadí vozidla z různých důvodů splývat s hlavou řidiče, bude potřeba Gaussových funkcí co nejvíce. Pokud je ovšem kontrast dostatečný, je možné vyobrazit si jednotlivé shluky samostatně, viz obrázek 4-6.



Obrázek 4-6: Rozdělení vstupního obrázku na jednotlivé shluky algoritmem GMM

Z obrázku 4-6 je pro detekci pochopitelně nejvíce zajímavý shluk, který vykresluje hlavu řidiče. Ten může po provedení morfologického uzavření vypadat následovně:



Obrázek 4-7: Provedení morfologického uzavření na shluk představující hlavu

Pro vyznačení oblasti hlavy je nejlepší použít algoritmus hledání kontur. Kontury musí být spojitě, zároveň je ale jisté, že jich ve vstupním obrázku bude možné najít mnoho. Aby bylo možné s jistotou tvrdit, že se jedná o konturu hlavy, je nutné srovnávat

velikost plochy, kterou opisuje s intervalem referenčních hodnot. Interval si lze představit jako rozsah hodnot ploch, které mohou zabírat hlavy řidičů v obrázku. Závisí to samozřejmě i na vzdálenosti od kamery a také rozlišení.



Obrázek 4-8: Výsledná detekce hlavy algoritmem GMM

Výsledkem detekce je určení oblasti hlavy ve vstupních obrázcích. To lze využít pro měření větších odchylek hlavy řidiče, v případě, že se řidič nakloní do jedné nebo druhé strany. Detekovanou oblast lze rovněž využít pro zúžení oblasti ve vstupních snímcích pro hledání očí i úst.

Detekce hlavy algoritmem GMM byla zkoušena na relativně malém datasetu, fungovala ovšem celkem spolehlivě. Bohužel, během testování se ukázalo, že délka provedení algoritmu je spíše v sekundách, než v milisekundách. Je otázkou, zdali lze rychlost algoritmu zvýšit na takovou úroveň, aby bylo možné kontrolovat polohu hlavy alespoň každou sekundu.

### **Detekce rukou**

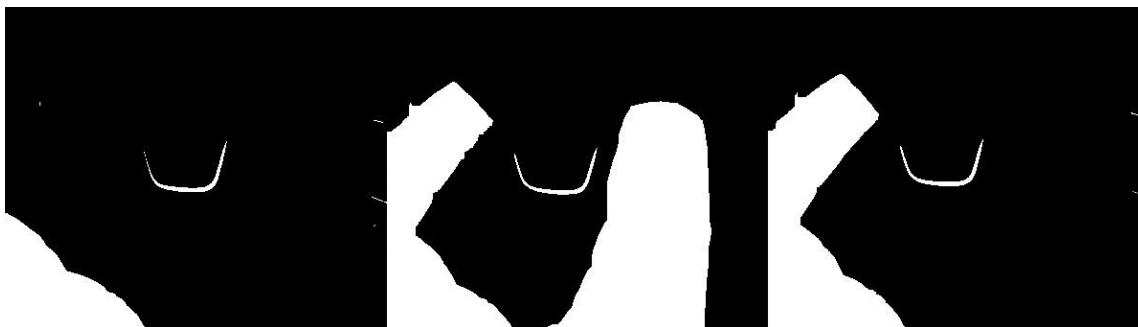
Algoritmus GMM lze teoreticky použít i pro detekci rukou řidiče na volantu vozidla. Zde by mohlo postačovat použití méně Gaussových funkcí, než u detekce hlavy řidiče. Pro zamyšlení se nad možným principem detekce rukou je nejlepší si nejprve vyobrazit snímky s rukama na volantu a s rukama mimo volant, a to po aplikaci algoritmu GMM.



Obrázek 4-9: Algoritmus GMM pro segmentaci rukou

Na obrázku 4-9 je možné vidět rozdělení shluků ve vstupních snímcích, které zobrazují situaci, kdy řidič drží volant dvěma rukama, jednou, nebo ho vůbec nedrží. Byly zde použity tři Gaussovy funkce. Hned na první pohled je zjevné, že dlaně vs. samotné ruce patří do jiných shluků. Závisí to od barvy bundy, která je v tomto případě červená. Pokud by řidič žádnou bundu neměl, celé ruce, včetně dlaní, by mohly, v závislosti na osvětlení, patřit do jednoho shluku. Naopak, pokud by řidič měl na sobě bundu tmavé barvy (podobně jako barva interiéru vozidla), pixely tvořící ruce by patřily do stejného shluku jako interiér vozidla.

Za těchto okolností nelze použití metody GMM pro detekci rukou na volantu považovat za příliš spolehlivou. Nicméně i při znalosti těchto limitů by šlo srovnávat plochy, které zabírají ruce, případně i tělo řidiče, pokud se nacházejí v záběru. Pokud řidič ruce na volantu má, je plocha, kterou tvoří shluk rukou velká. Pokud řidič drží volant jednou rukou, je plocha pořád větší, než když volant nedrží vůbec.



Obrázek 4-10: Možné použití algoritmu GMM pro detekci rukou na volantu

Algoritmus GMM by bylo možné otestovat i u systému, který kontroluje přítomnost nohou u pedálů. Platily by zde podobné limity jako u detekce rukou.

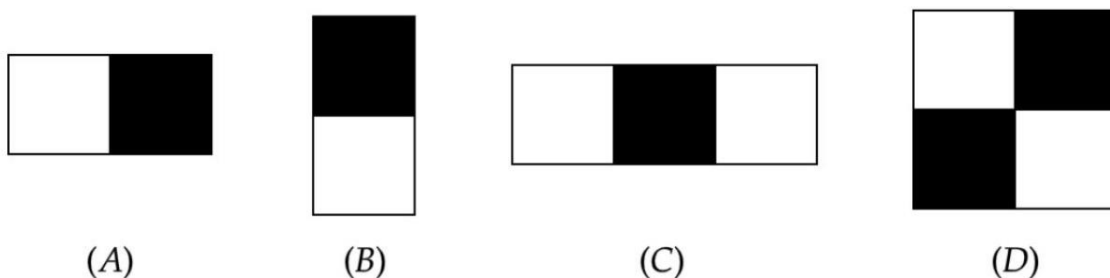
#### 4.2.2 Detekce metodou Haarovy kaskády

Jiným typem detekce je detekce objektů použitím Haarovy kaskády. Jedná se o segmentační metodu navrženou v roce 2001 Paulem Violou a Michaellem Jonesem. Je to metoda založená na strojovém učení. Používá tzv. kaskádní funkci, která je

natrénována z velkého množství referenčních obrázků. Ty jsou buď pozitivní, to znamená, že obsahují požadovaný objekt detekce, nebo jsou negativní, neobsahují objekt detekce. Kaskádní funkce je poté použita pro detekci hledaného objektu, v případě této práce očí, resp. úst, v jiných obrázcích. Pro bližší pochopení této metody je potřeba se zamyslet nad následujícími pojmy [28].

## Haarovy příznaky

Haarovy příznaky si lze představit jako matice jasových hodnot, ve kterých je část obrazu čistě bílá a část čistě černá. Tyto matice dobře ilustruje obrázek 4-11.



Obrázek 4-11: Vybrané typy Haarových příznaků [28]

Každý Haarův příznak lze popsat jedním číslem odečtením sumy jasových hodnot černých regionů, od sumy jasových hodnot bílých regionů. U ideálních modelů Haarových příznaků by se jednalo o hodnotu 255.

Při hledání těchto příznaků v reálných obrázcích, ale nejspíše není možné najít jejich ideální tvar. Využívá se předpokladu, který mluví o tom, že čím vyšší je výsledná hodnota popisující možný Haarův příznak, tím pravděpodobnější je, že byl příznak nalezen.

Je také nutné uvědomit si, že příznaků může být při využití všech velikostí matic a všech poloh velmi velké množství. Tím pádem jsou na výpočetní výkon kladeny velmi vysoké požadavky. Tento problém pomáhá řešit tzv. integrální obraz.

## Integrální obraz

Integrální obraz je takový pohled na obraz, ve kterém je hodnota bodu v oblasti se souřadnicemi  $[x, y]$  rovná sumě pixelů nad a vlevo od daného bodu (včetně) v standardním obrazu.

Obrázek				Integrální obrázek			
0	1	1	1	0	1	2	3
1	2	2	3	1	4	7	11
1	2	1	1	2	7	11	16
1	3	1	0	3	11	16	21

Obrázek 4-12: Převedení vstupního obrázku na integrální obrázek [29]

Na základě této znalosti lze hodnotu součtu libovolného prostoru v obrazu potřebnou pro nalezení Haarových příznaků získat prakticky ihned, bez neustálého počítání.

## Adaboost

Pravdou je, že velká většina příznaků nemá pro účel detekce velkou relevanci. Otázkou je, jak jejich počet rozumně snížit a zároveň zachovat ty, které jsou potřebné pro natrénování klasifikátoru pro detekci hledaného objektu. To je možné řešit pomocí algoritmu AdaBoost. Jedná se o algoritmus strojového učení, který je určen pro výběr nejlepších příznaků ze všech možných.

Algoritmus funguje tak, že aplikuje každý z možných příznaků na jednotlivé obrázky z referenčního datasetu. Ten je rozdělen na dvě skupiny, kde v první se nachází pouze obrázky, které obsahují hledaný objekt. Ve druhé skupině jsou obrázky, které neobsahují hledaný objekt. Každý z aplikovaných příznaků má za úkol klasifikovat jednotlivé obrázky. V prvním kroku je každému z obrázku přiřazena stejná váha. Nalezne se příznak s největší úspěšností, tzv. slabý klasifikátor. Na základě jeho chybovosti se vypočte váha nalezeného klasifikátoru.

$$\text{Váha klasifikátoru} = \frac{1}{2} \log \left( \frac{1 - \text{celková chybovost}}{\text{celková chybovost}} \right) \quad (2)$$

Celková chybovost je poměr špatně detekovaných obrázků k celkovému počtu obrázků v datasetu.

Aby bylo možné vzít v potaz vlastnosti klasifikátoru z první iterace, v další iteraci se změní váhy jednotlivých obrázků s důrazem na obrázky, které první klasifikátor detekoval špatně. Postupuje se podle tohoto vzorce:



$$\text{Nová váha obrázku} = \text{Váha obrázku} * e^{\text{váha klasifikátoru}} \quad (3)$$

Ze vzorce je možné vytušit, že pokud je váha klasifikátoru malá, váha špatně detekovaných obrázků se změní pouze nepatrně. Pokud je naopak váha klasifikátoru velká, váha chybně detekovaných obrázků se rapidně zvýší. Váhy správně detekovaných obrázků se naopak zmenší podle podobného vzorce, kde jediný rozdíl představuje negativní koeficient Eulerova čísla.

$$\text{Nová váha obrázku} = \text{Váha obrázku} * e^{-\text{váha klasifikátoru}} \quad (4)$$

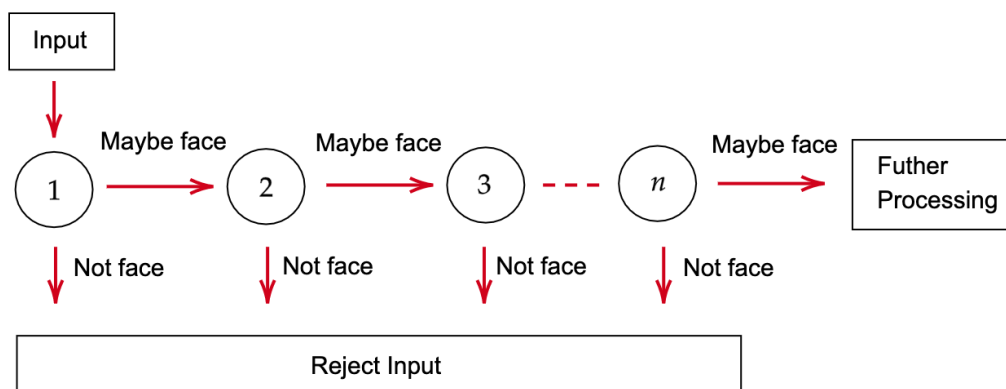
Pokud je váha klasifikátoru z předešlé iterace velká, nová váha správně detekovaných obrázků se razantně zmenší.

Nakonec se váhy obrázků normalizují tak, aby byl jejich součet roven jedné. Postup se v další iteraci opakuje a vybere se další slabý klasifikátor. Iterační proces probíhá tolikrát, kolik slabých klasifikátorů se požaduje. Výsledný silný klasifikátor je složen z jednotlivých slabých klasifikátorů, které jsou vážené určenými váhami, viz vzorec níže [30].

$$F(x) = w_1 f_1(x) + w_2 f_2(x) + w_3 f_3(x) + \dots \quad (5)$$

## Princip kaskády

Po aplikování algoritmu AdaBoost může počet používaných příznaků klesnout i stonásobně. I tak jich ale zůstává relativně velký počet. Pro další šetření výpočetního výkonu lze použít princip kaskády. Ta rozděljuje vybrané slabé klasifikátory do několika úrovní, každá úroveň tvoří silný klasifikátor. S každou úrovní počet slabých klasifikátorů stoupá. Při analýze obrazu se postupuje tak, že se na daném okně v obrazu postupně zkouší jednotlivé úrovně – silné klasifikátory. Pokud, některý z nich vyše negativní výstup, v analýze pomocí dalších klasifikátorů v pořadí se již nepokračuje. Místo toho se analýza soustředí na nové okno v obrazu. Tímto způsobem je možné se rychle zbavit oblastí v obrazu, které neobsahují hledaný objekt. Názorný příklad funkce kaskádní klasifikace je na obrázku 4-13.



Obrázek 4-13: Kaskádní klasifikátor [28]

## Testování metody Haarovy kaskády

I když lze Haarovou kaskádu využít pro jakékoliv hledané objekty (pokud se povede úspěšně natrénovat neuronovou síť), pro případ této práce najde uplatnění především při detekci obličeje nebo očí.

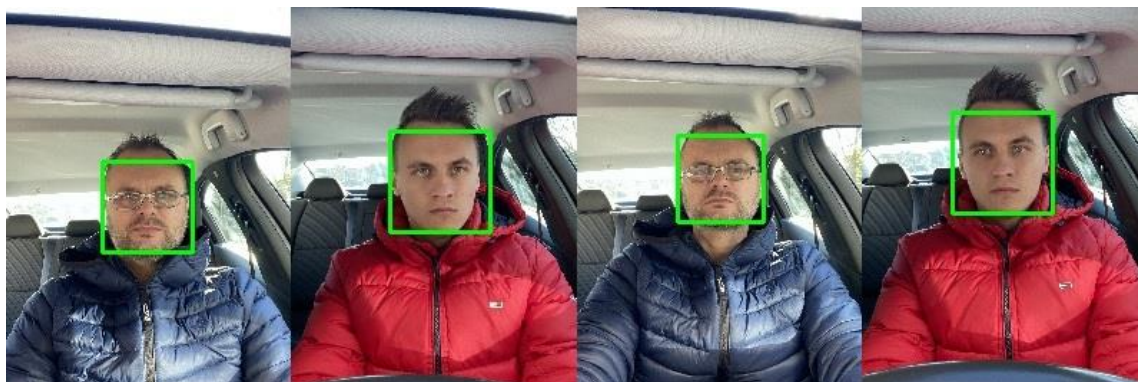
### Detekce obličeje

K otestování této metody pro detekci obličeje je možné využít natrénovanou neuronovou síť z knihovny *OpenCV*, konkrétně *haarcascade\_frontalface\_default*. Tato síť je typu 24x24.

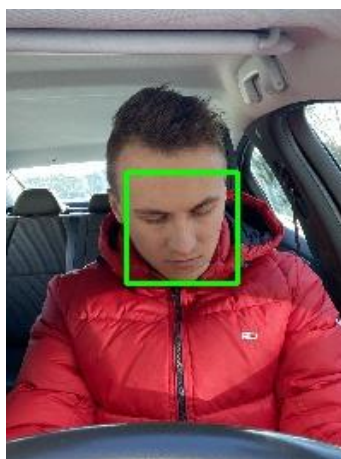
Při testování se osvědčilo obrázky před detekcí předzpracovat. Přesněji se jednalo o potlačení šumu a provedení operace morfologického uzavření. Tyto kroky snížily míru falešných detekcí.

Klasifikátor, vycházející ze svého principu, vrátí souřadnice čtyř bodů tvořících čtverec. Na základě této znalosti lze teoreticky měřit odchylku hlavy od požadované polohy. Tuto metodu lze tím pádem použít pro systém, který řeší připravenost řidiče převzít řízení. Vyčlenění oblasti obličeje v obrázku by šlo rovněž použít jako základ pro detekci očí resp. úst. Pro oči nebo ústa je ovšem taky možné použít metodu Haarovy kaskády. Vymezení oblasti je tudíž vhodné především za předpokladu využití jiných metod detekce očí nebo úst.

Ve výsledku funguje detekce obličeje metodou Haarovy kaskády velice spolehlivě a samotná metoda je považována za jakýsi zlatý standard detekce tváří v oboru počítačového vidění. Detekci tváří na testovaných snímcích lze vidět na obrázku 4-14. Je nutno podotknout, že tento typ detekce není určen pro natočení hlavy do stran, jelikož v takových případech některé části obličeje, důležité pro detekci, není na snímku vidět. Při natočení hlavy pohledem dolu nebo vzhůru detekce funguje, zde zůstává pohled na obě oči, nos apod. zachován, viz obrázek 4-15.



Obrázek 4-14: Detekce obličeje metodou Haarovy kaskády

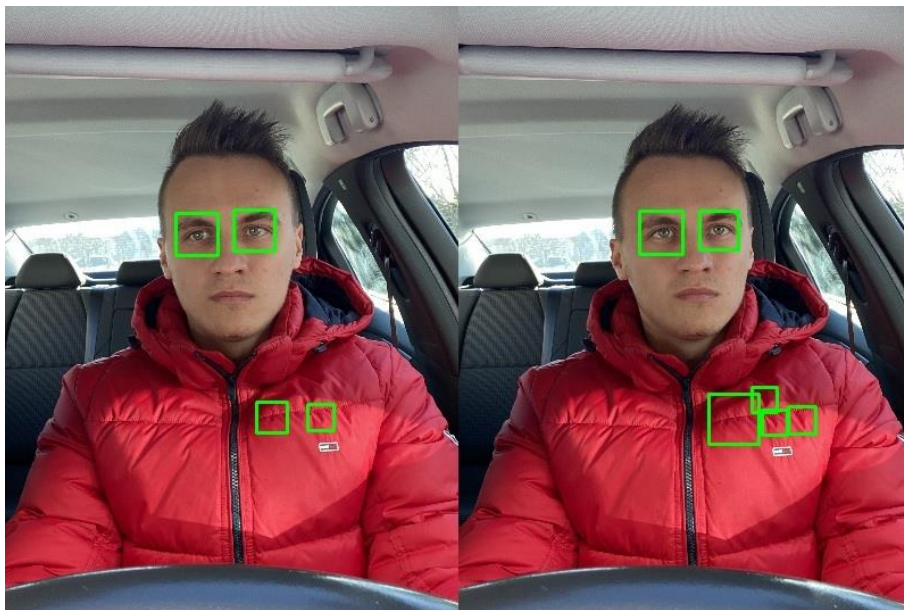


Obrázek 4-15: Detekce obličeje metodou Haarovy kaskády – pohled dolu

### Detekce očí

Při otestování Haarovy kaskády pro detekci očí je opět možné vycházet z natrénované neuronové sítě od knihovny *OpenCV*, tentokrát s názvem *haarcascade\_eye*. Platí zde podobné principy předzpracování, jako u detekce obličeje.

Výsledky Haarovy kaskády pro detekci očí již nevykazují tak spolehlivou míru detekce, jako tomu bylo u detekce obličeje. I když kaskáda u testovaných snímků pokaždé detekuje obě oči, spolu se správnou detekcí očí, ale vykazuje i značnou míru falešných detekcí – obrázek 4-16. To by šlo řešit několika způsoby. Buď by se kamera mohla soustředit pouze na oblast předpokládaného výskytu řidičovy hlavy, nebo by šlo srovnávat obsahy jednotlivých čtverců – oblast oka, s definovanou referenční hodnotou. Je možné, že míru falešné detekce lze ovlivnit i správnou kombinací kroků předzpracování obrazu.



Obrázek 4-16: Detekce očí metodou Haarovy kaskády

### 4.2.3 Hranové operátory pro detekci mrkání

Využitím Haarovy kaskády pro vymezení oblastí, kde se nachází oči řidiče, je dále možné detekovat samotný obrys oka. Ten je zajímavý v kontextu plně otevřeného nebo plně zavřeného oka. Modelové snímky pro plně otevřené a plně zavřené oko jsou na obrázku 4-17.



Obrázek 4-17: Plně otevřené oko řidiče vs. plně zavřené oko řidiče

Aby bylo možné získat obrys oka, využije se tzv. hranový operátor. Jedná se o matici, obvykle o rozměru 3x3, která se operátorem konvoluce aplikuje na vstupní snímek. Místa, která obsahují velké jasové změny, se ve výstupním snímku projeví jako hrany. Mezi nejefektivnější hranové operátory patří tzv. Cannyho operátor. Jeho základem je Sobelův operátor, navíc ale využívá proces hystereze. Ta spočívá v tom, že uživatel zadá interval, který tvoří spodní a horní práh. Všechny hranové pixely pod prahem jsou potlačeny, všechny pixely nad prahem jsou automaticky akceptovány jako hrany. Pixely v intervalu jsou ponechány, pouze pokud jsou spojeny s pixely s hodnotami nad prahem [31].

## Testování metody hranových operátorů

Po mírném předzpracování vstupních snímků očí je možné obdržet následovné obrázky hran:



Obrázek 4-18: Obráz hran u plně otevřeného vs. plně zavřeného oka

Z obrazů hran lze dedukovat, že počet hran v obrazu plně otevřeného oka je zhruba 2x větší, než počet hran u plně zavřeného oka (lze vypočítat pomocí histogramu). Na základě tohoto předpokladu by šlo učinit rozhodnutí, kdy je oko otevřeno a kdy zavřeno. Metoda je ale velice citlivá na světelné podmínky a také správnou kombinaci předzpracování a nelze ji považovat za spolehlivou.

Pro zajímavost je možné na obrázky očí vyzkoušet algoritmus GMM.



Obrázek 4-19: Obráz hran u plně otevřeného vs. plně zavřeného oka u metody GMM

Plocha shluku u otevřeného oka je mnohem větší, než u oka zavřeného. I touto metodou by šlo detekovat mrkání.

### 4.2.4 Zhodnocení navržených metod pro 2D snímky

V rámci prvotních pokusů byly představeny a realizovány tři metody segmentace, resp. detekce různých žádaných částí těla člověka. Metod segmentace, které lze pro 2D snímky použít je samozřejmě hodně, v rámci pokusů byly ale vybrány pouze ty, u kterých byl větší předpoklad úspěšnosti.

U první z metod, metody Gaussian mixture model, se podle očekávání projevil nedostatek 2D snímků a jejich nevhodnost pro potřeby úlohy segmentace řidiče ve vozidle. I když tuto metodu nelze považovat za úplně nespolehlivou a u řidiče s kontrastním oblečením fungovala poměrně dobře, je to právě podmínka nutnosti kontrastu oproti pozadí, která dělá tuto metodu nevhodnou pro tvorbu funkčních navrhovaných systémů. Dalším nedostatkem se ukázala být velká výpočetní náročnost

algoritmu. Jedna iterace provedení algoritmu GMM trvala několik sekund, což není pro real-time hlídání pohybů řidiče akceptovatelné.

Detekce metodou Haarovy kaskády se jeví jako o něco perspektivnější metoda. Algoritmy založené na strojovém učení jsou známé svou dobrou spolehlivostí detekce a tento fakt se potvrdil i při testování metod pro detekci obličeje a očí řidiče. Oproti metodě GMM je rychlost provedení jedné iterace algoritmu Haarovy kaskády řádově vyšší, tudíž lze očekávat možnost použití při real-time detekci.

Není pravděpodobné, že by stereo snímky a tedy dodatečná informace o hloubce v obrazu otevřela možnosti pro lepší algoritmy detekce očí nebo úst, lze proto konstatovat, že s Haarovou kaskádou nebo podobnou metodou založenou na strojovém učení je možné v navrhnutém systému pro detekci ospalosti počítat.

Na metodu Haarovy kaskády bylo v rámci pokusných metod navázáno použití hranových operátorů. Ty měly za úkol rozeznat rozdíl mezi otevřeným nebo zavřeným okem (analogicky by šlo použít i pro ústa). Spolehlivost rozeznávání těchto dvou stavů na základě počtu hran sice teoreticky funguje, nicméně citlivost na světelné podmínky a další parametry dělá metodu hranových operátorů nevhodnou pro použití ve výsledném systému. Pro spolehlivé určení mrkání a zívání bude potřeba detekovat oči, resp. ústa ve formě charakteristických bodů a poté pracovat s hodnotami poměru, který je popsán studií zmíněné v dřívější podkapitole s názvem *Detekce ospalosti založená na hlídání očí řidiče*. Níže je statistický přehled procentuální úspěšnosti jednotlivých metod, které umožňovaly real-time snímání, testovaných na videu o délce 30 s. Výsledky jsou závislé na světelných podmínkách, natočení hlavy řidiče a pozadí. Nesprávná detekce je každá, která detekuje i jiné, než požadované části/stav, nebo je nenalezne vůbec.

Typ metody	Správná detekce	Nesprávná detekce
Haarova kaskáda - obličej	85,32%	14,68%
Haarova kaskáda - oči	65,77%	34,23%
Hranové operátory - oči	51,51%	48,49%

### 4.3 Rozbor, návrh a realizace metod - vstupní snímky ze stereo kamery

Snímky pořízené stereo kamerou poskytují oproti běžným dvourozměrným jednu velkou výhodu. Na základě principu stereovidění je možné z těchto snímků pro každý bod extrahovat informaci o prostorové souřadnici osy „z“. Společně tyto souřadnice tvoří tzv. hloubkovou mapu. Platí, že segmentace obrazu je proces, který má za úkol oddělit popředí (objekt) od pozadí, jinak řečeno oddělit věci, které jsou blíže ke kameře od věcí, které jsou od kamery daleko. A to je právě to, co prostorové souřadnice popisují. Tím

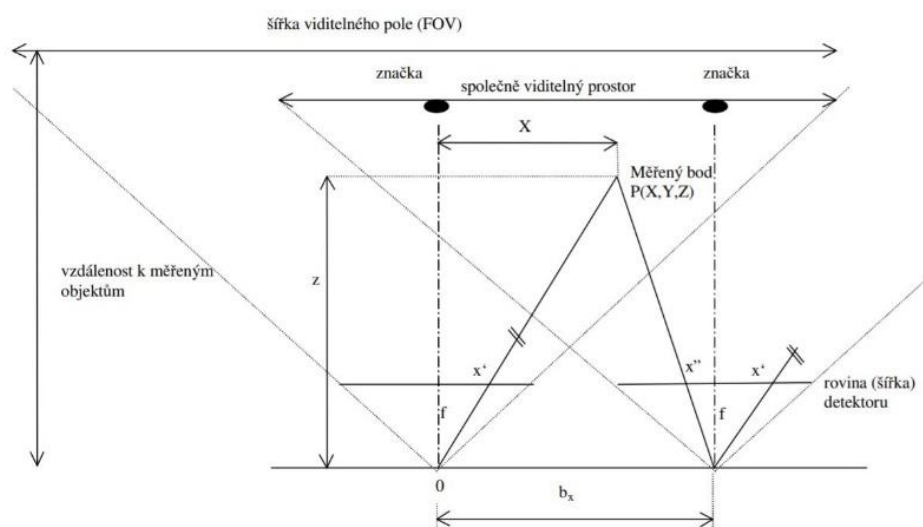
pádem není nutné spoléhat se na homogenitu pixelů ve smyslu jasové úrovně, naopak postačí najít vhodný práh a k označení segmentovaného objektu využít hloubkovou mapu.

Segmentace objektů na základě stereovidění má díky své jednoduchosti a zároveň invarianci vůči rozličným světelným podmínkám velký potenciál spolehlivě segmentovat řidiče, resp. hledané části jeho těla. Je tedy předpoklad, že právě tento princip bude možné úspěšně aplikovat do algoritmů navrhovaných systémů. Nejprve je však dobré vysvětlit, v čem spočívá princip stereo kamer, resp. výpočet prostorových souřadnic.

### 4.3.1 Princip stereovidění

Princip stereovidění spočívá v technice pasivní triangulace. Jedná se o fotogrammetrickou rekonstrukci snímaného objektu na základě jeho průmětů na rovinu snímacího zařízení bez přídavného zdroje světla. Projekcí se ztrácí jeden rozměr, který je ovšem možné vypočítat, pokud je snímáný objekt nasnímán pomocí více kamer [32]. Tyto kamery snímají objekt z různých pohledů, musí ale platit, že se zmíněný objekt nachází v zorném poli obou snímacích zařízení. Pro standardní aplikaci stereovidění se počítá s kamerami s rovnoběžnými optickými osami, jak je to i v případě kamery Intel RealSense. V takovém případě není v podstatě nutné počítat vnější parametry kamer, které jsou pro každé umístění kamer jiné.

Výpočet třetího rozměru pomocí triangulace je založen na geometrických vlastnostech trojúhelníku. Na obrázku 4-20 je první a druhý vrchol triangulačního trojúhelníku tvořeny středy kamer, třetí vrchol představuje měřený bod P. Strany trojúhelníku, které tvoří měřený bod s každým z optických středů, jsou průměty bodu na jednotlivé obrazové roviny. Strana tvořená optickými středy představuje triangulační bázi.



Obrázek 4-20: Model stereo kamery [32]



Pokud je známá triangulační báze - vzdálenost mezi středy kamer a ohnisko (vnitřní parametry kamery, které lze vyčíst z dokumentace), a zároveň jsou známy úhly, které svírá měřený bod s obrazovými rovinami obou kamer, lze využít následující výpočetní vztahy:

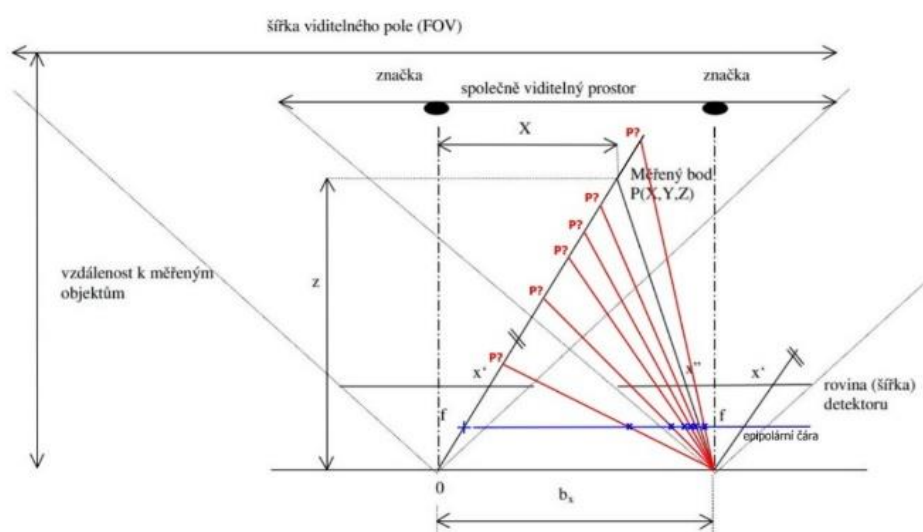
$$X = x' \frac{b_x}{p} \quad Y = y' \frac{b_x}{p} \quad Z = f \frac{b_x}{p}, \text{ kde } p = x' - x'' \quad (6)$$

Platí, že proměnné  $x'$  a  $y'$  jsou průměty bodu  $P$  na snímáček první kamery a  $b_x$  je velikost báze. Proměnná  $f$  představuje ohnisko, které je v ideálním případě u obou kamer stejné. Poslední proměnná představuje tzv. paralaxu. Jedná se o rozdíl v pozicích analyzovaného bodu, který je projektován na obě z kamer. Tyto vztahy popisují skutečné souřadnice (v metrech) bodu  $P$  a je to právě souřadnice  $Z$ , která určuje reálnou vzdálenost snímaného objektu od kamery.

## Problém korespondence

Stereovidění obnáší jeden významný problém. Je to problém korespondence, který řeší, jak určit, kde se analyzovaný bod v prostoru zobrazil, a to jak na jedné, tak druhé kameře. Bez této informace nejsou známy úhly triangulačního trojúhelníku a není možné vypočítat souřadnici  $Z$ .

Společné dvojice bodů je možné hledat moderními algoritmy, jako jsou SIFT nebo SURF, které detekují a popisují významné body v obraze. I když se o tom dokumentace explicitně nezmiňuje, i kamera Intel RealSense pravděpodobně pro řešení problému korespondence používá právě tyto metody, které zpracovává procesor kamery Vision Procesor D4.



Obrázek 4-21: Epipolární geometrie a zjednodušení problému korespondence



Pokud by ale bylo potřeba hledat jednotlivé páry z celé množiny nasnímaných bodů, jednalo by se o výpočetně velice náročný úkol. Naštěstí lze tuto úlohu zjednodušit na základě poznatků z epipolární geometrie.

Epipolární geometrie popisuje geometrické zákonitosti, které se uplatňují při snímání jedné scény z dvou různých pohledů. Dobře to vysvětluje upravený obrázek 4-21 výše.

Pokud se analyzuje konkrétní bod, který se ve formě pixelu zobrazil na kameře vlevo, nelze předem určit, jak daleko se od kamery nachází. Při sestavování triangulačního trojúhelníku se poloha jeho strany, která určuje průmět hledaného bodu na kameře č. 2, může měnit, a to v závislosti na jeho vzdálenosti od kamery. V obrázku tento jev představují vyznačené červené čáry. Všechny teoreticky možné průměty červených čar s obrazovou rovinou tím pádem určují, kde všude by se pozorovaný bod na druhé kameře mohl zobrazit. Tyto průměty tvoří tzv. epipolární čáru. Ta je, v případě kamer s rovnoběžnými optickými osami, rovnoběžná s obrazovou rovinou snímáče [33].

Díky této zákonitosti postačuje při řešení problému korespondence porovnat analyzovaný pixel pouze s pixely ležící na epipolární čáře - výpočetní náročnost problému řádově klesá.

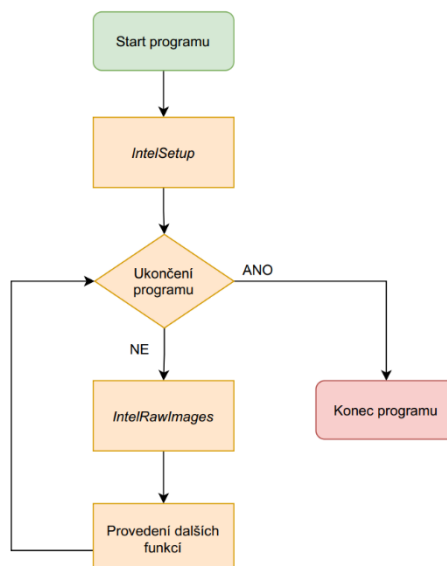
Pro další zjednodušení problému korespondence se používá algoritmus rektifikace obrazu. Jedná se o geometrickou transformaci, která má za úkol zabezpečit, aby všechny korespondenční body ležely na jedné přímce. To se uplatní především u kamer, které nejsou v poloze rovnoběžných optických os. Je patrné, že to není případ kamery Intel RealSense, ale i u takových kamer se obvykle rektifikace provádí z důvodu drobných odchylek poloh objektivů, které mohou v průběhu života kamery nastat.

### 4.3.2 Funkce pro segmentaci na základě hloubky objektů v obrazu

Po vysvětlení principu stereovidění je možné přesunout se k aplikační části této práce pro 3D snímky. První funkcionalitu, kterou je potřeba navrhnout je dozajista řešení, které umožní segmentovat požadované objekty na základě jejich vzdálenosti od kamery a využije se jako základ pro všechny navrhované systémy.

Pro vytvoření této funkcionality je nutné umět pracovat s kamerou přes zvolený jazyk python. To usnadňuje wrapper s názvem *pyrealsense2*, který poskytuje vazbu jazyka python na jednotlivé funkce Intel SDK.

Na základě těchto předpokladů se lze pustit do vytváření zmíněného řešení. Pro přehlednost je vhodné ho rozdělit do dvou funkcí. První funkce bude obsluhovat obecný setup kamery a postačí ji zavolat pouze při startu programu. Druhá bude řešit nalinkování obrazů z obou kanálů a následnou segmentaci. Tato funkce i všechny další vytvářené funkce budou umístěny ve while smyčce, která se bude opakovat do doby, než se nevypne program. Základní kostru programu si lze představit takto:



Obrázek 4-22: Základní blokové schéma programu

V prvním kroku první funkce s názvem *IntelSetup* je vždy nutné vytvořit objekt typu pipeline a povolit pomocí funkce *enable\_stream* snímání z obou kanálů kamery. Zároveň je v rámci konfigurace nutné zvolit rozlišení, kterým bude každý z kanálů snímat. Dle dokumentace kamery, která je rozebrána i v podkapitole 3.2, je hloubkový kanál schopen snímat v rozlišení, které odpovídá HD, RGB kanál dokonce ve full-HD. S vyšším rozlišením, bohužel, neúměrně rostou nároky na výpočetní výkon, což představuje problém především při real-time snímání a vyhodnocování. Navrhované systémy, ale nepožadují příliš vysoké rozlišení vstupních snímků, postačí tedy použít rozlišení typu SD – 480x640.

Aby bylo možné segmentovat objekty na základě reálné vzdálenosti od kamery v metrech, je dále potřeba zjistit převodní vztah mezi skutečnou vzdáleností a hloubkovým obrazem, který kamera snímá. Ten je možné určit z funkce *get\_depth\_scale*, která obsahuje údaj o hloubkovém měřítku kamery. Funkce pro kameru D415 vrátí hodnotu 0,001. To znamená, že například hodnota 1000 v hloubkovém obrazu představuje pro zvolený bod jeden metr od kamery. Převodní vztah pro bod se souřadnicemi  $x, y$  v hloubkovém obraze tedy vypadá následovně:

$$vzdálenost\ v\ metrech = hloubkový\ obraz\ [x, y] * 0,001 \quad (7)$$

V dalším kroku tvorby funkce je nutné nadefinovat typ linkování obrazů z obou kanálů kamery, tedy jak hloubkového, tak RGB kanálu. To zabezpečí funkce *align*, která jako parametr přijímá typ kanálu, na který je potřeba namapovat druhý typ obrazu. Pro případ funkce segmentace je vhodné namapovat hloubkový obraz na obraz typu RGB.

Po zadefinování potřebných předpokladů pro snímání a segmentaci je funkce *IntelSetup* hotová.

Dále je možné se pustit do vytvoření funkce pro samotnou segmentaci – *IntelRawImages*. Ta si vždycky nejprve vezme aktuální snímky z obou kanálů přes funkci *wait\_for\_frames*, a poté musí provést samotné namapování. Zde je možné narazit na problém, který se týká odlišné hodnoty horizontálního úhlu zorného pole u jednotlivých modulů kamery. Dokumentace uvádí, že tato hodnota je u hloubkového modulu o pár stupňů menší, než u RGB modulu. Hloubkový snímek je tudíž v ose X o něco menší (zhruba o 100 pixelů) a je nevyhnutné to brát při vytváření detekčních algoritmů do úvahy. Pro doplnění je nutno dodat, že pomocí těchto funkcí byly nasnímány vstupní snímky, které byly rozebrány v podkapitole 3.4 *Nasnímání testovacích snímků*.

Po splnění popisovaných kroků je již možné segmentovat. Segmentace se uskutečňuje na základě jednoduché podmínky. Všechny body z RGB obrazu se zobrazí v segmentovaném obrazu pouze tehdy, je-li hodnota jejich příslouchajících bodů v hloubkovém obrazu větší, než hodnota zvolené vzdálenosti od kamery, na jejíž základě má segmentace proběhnout. Pokud některý bod tuto podmínku nesplňuje, ve výstupním snímku se mu přiřadí hodnota nula.

### **Testování detekce na základě hloubky v obrazu (měřicí scéna č. 1)**

Prvotní testování vytvářené funkcionality probíhalo snímáním autora této práce sedícího na židli u počítače, která měla představovat sedadlo automobilu. Po zvolení vhodného prahu segmentace byla funkčnost metody pro tuto scénu zcela bezchybná. Problém nepředstavovaly zhoršené světelné podmínky, ani úplná tma. Rovněž nezáleželo na typu pozadí, resp. jeho kontrastu se segmentovanou osobou. Testováním metody se tedy potvrdilo, že princip segmentace na základě hloubky je vůči těmto vlivům zcela invariantní a rozhodně vhodný pro použití. Výsledek je vyobrazen na obrázku níže v levé části.



Obrázek 4-23: Testování segmentace na základě hloubky v obrazu

Při testování algoritmu v rámci měřicí scény č. 1 se situace mírně zkomplikovala. Problém je možné pozorovat na stejném obrázku vpravo, kde se při segmentaci nepovedlo zbavit malé části pozadí. Konkrétně se jedná o kousek stropu, který tvoří spojovací část mezi A a B-sloupkem vozidla. Také je možné si v segmentovaném obrázku všimnout velmi malé části řidičových dveří. Tyto nedokonalosti jsou způsobeny vzdáleností těchto částí pozadí od kamery, která je menší než stanovený práh.

Na první pohled je řešení snadné, postačí zmenšit délku prahu segmentace tak, aby nepokrýval žádnou část pozadí. Bohužel, v takovém případě sice dojde k oddělení celého pozadí, ale zároveň se tím ztratí části řidičova těla, které jsou ve stejné vzdálenosti od kamery, jako problematické úseky pozadí. Touto cestou tím pádem nelze jít, hodnota prahu segmentace musí být vždycky větší než vzdálenost řidiče od kamery.

Tento nedostatek se nemusí v rámci měřicí scény projevit u každého typu vozidla stejně. U vozidel s vyšší střechou, např. typ SUV je pravděpodobná segmentace bez jakýchkoliv problémů. Částečně je ovšem možné tuto komplikaci řešit jiným postavením kamery tak, aby měřicí scéna neobsahovala problematické kousky pozadí. To ale způsobí, že se řidič nebude nacházet ve středu snímané plochy, což by pokazilo návrh algoritmů pro analýzu pohybů řidiče. Zůstává tedy tento nedostatek segmentace vzít na vědomí a při vytváření navazujících algoritmů s ním počítat.

Na základě těchto zkušeností lze zhodnotit, že aby byla segmentace úspěšná a bezchybná, musí všechny hodnoty souřadnic typu Z segmentovaného objektu patřit do intervalu určeného zvoleným prahem a zároveň zde nesmí patřit souřadnice typu Z pozadí. Pokud takový práh zvolit nelze, ve výstupním snímku se může část pozadí objevit.

Funkcionalitu segmentace na základě rozdílné hloubky v obraze lze nepochybně aplikovat i na měřicí scénu č. 2 a č. 3. Využití segmentace pro tyto scény se budou věnovat podkapitoly, které řeší algoritmy pro detekci držení volantů a detekci připravenosti stlačit pedály.

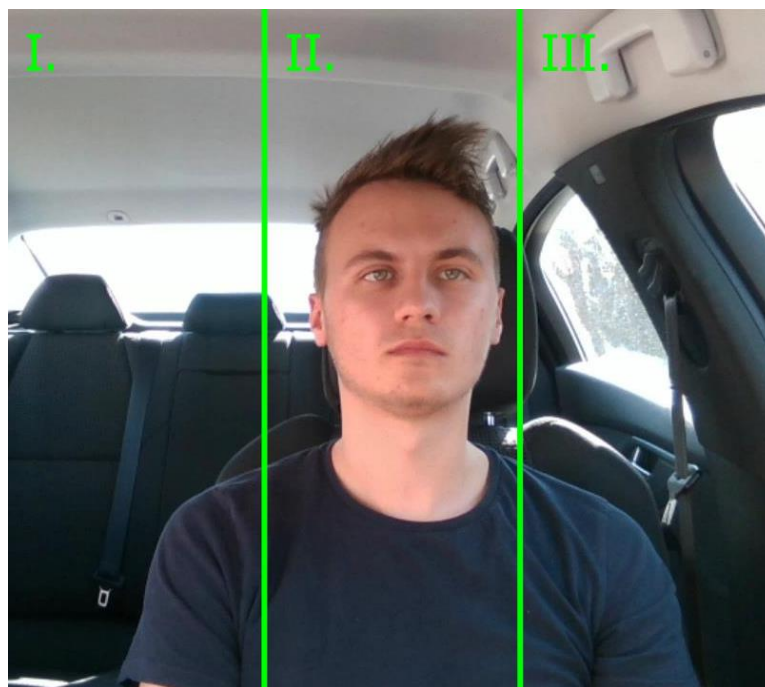
### **4.3.3 Detekce náklonu řidiče do jednotlivých stran**

Poté co se segmentace na základě hloubky v obraze ukázala jako velice spolehlivá, lze přistoupit k vytvoření algoritmů pro detekci náklonu řidiče do jednotlivých možných stran. Algoritmus by měl být schopen detekovat náklon řidiče dopředu, doleva, doprava a nakonec také dovrchu. Detekci jednotlivých náklonů je možné, jak bylo vysvětleno v základním rozboru systémů (podkapitola 4.1), určovat na základě pohybů hlavy řidiče a měření odchylek od referenční polohy hlavy. V prvním kroku je tím pádem nutné zamyslet se nad tím, co vlastně referenční poloha hlavy je a vytvořit funkci, která extrahuje data popisující tuto polohu pro srovnání s aktuální polohou.

## Změření referenční polohy

Za referenční polohu hlavy řidiče se dá považovat taková poloha, ve které se řidič vyskytuje většinu svého času a je bezpečná pro řízení vozidla. To znamená, že má tělo i hlavu ve vzpřímené poloze a hlavou se dívá na čelní sklo, resp. vozovku. V rámci detekčních algoritmů je tudíž potřeba takovou polohu zaznamenat a její parametry změřit. Navrhované algoritmy potřebují s referenční polohou pracovat od samotného začátku každé jízdy, proto měření referenční polohy musí proběhnout těsně před začátkem jízdy. V praxi by se měření mohlo spustit například stlačením tlačítka start pro nastartování vozidla. Po započetí měření se řidič musí nacházet v jeho obvyklé poloze. Měření se vykonává jako průměr dat z několika prvních snímků z kamery.

Samozřejmě vyvstává otázka, jak zabezpečit, aby se řidič v požadované poloze opravdu nacházel. Jako nejpravděpodobnější řešení se jeví detekce lidských očí v pevně daném oknu snímané plochy. Podmínka přítomnosti řidiče může být sestavena takto: Pokud se měřicí scéna č. 1 vertikálně rozdělí na tři stejně velké oblasti, musí být oči detekovány ve druhé, resp. středové oblasti. Situaci dobře popisuje obrázek 4-24. Detekce očí může využívat princip Haarovy kaskády, případně jiný postup. Problematikou detekce očí se bude dále zabývat podkapitola *Detekce mrkání a zívání*.



Obrázek 4-24: Stanovení referenční polohy hlavy pro měření

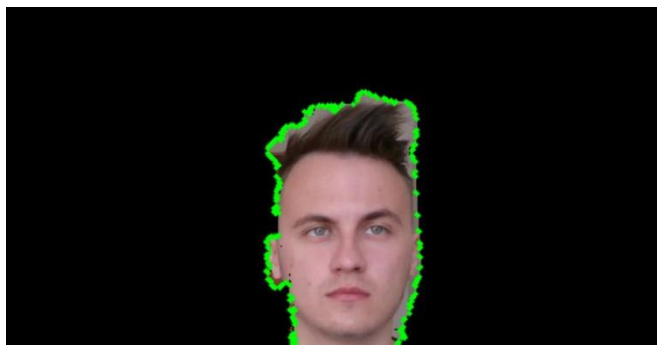
Pro extrakci dat z měření referenční polohy je vhodné vytvořit samostatnou funkci. Funkce se může jmenovat třeba *DetectionSetupScene1*. Pro svou činnost potřebuje snímky s vysegmentovaným řidičem, v rámci programu je proto vhodné

umístit ji hned pod funkci *IntelRawImages*, která zmíněné snímky vrací. Po provedení funkce se na kameře zobrazí věta „**Setup proveden**“.

Pro popis polohy hlavy je opět nejlepší využít princip knihovni funkce pro hledání kontur v obrazu. Ta vrací souřadnice jednotlivých bodů, které tvoří nalezené hrany v obrazu. Vstupní snímek funkce *DetectionSetupScene1* je ale tvořen celým řidičem (bez pozadí), je proto potřeba najít způsob, jak vyznačit konturou pouze oblast hlavy. Zcela nejjednodušším řešením je hledat hlavu pouze v horní polovině snímku rozděleného v horizontálním smyslu. Tento předpoklad je možné považovat za spolehlivý, jelikož u všech pohybů, které chce algoritmus detekce pohybů řidiče detekovat, bude hlava řidiče, nebo alespoň její část vždycky v horní polovině obrazu.

Funkce po tomto zjednodušení vrátí všechny kontury, které se zde nachází. Ve snímku s odstraněným pozadím se jedná pouze o konturu, která popisuje obálku hlavy. Pokud se pozadí nepovede úplně odstranit, funkce vrátí kontur více. Pro takový případ je vhodné vytvořit pomocnou funkci *GetLargestContour*, která nalezne a setřídí kontury od největší po nejmenší. Předpokládá se, že je to právě největší z kontur, která popisuje hledanou hlavu. V případě špatně odděleného pozadí ovšem tohle nemusí platit, v takové situaci je nutno zamyslet se nad jiným umístěním kamery.

Po nalezení kontury obálky hlavy je k dispozici informace (souřadnice) o aktuální poloze hlavy. Pokud se tyto data zprůměrují z několika prvních snímků, obdrží se informace o požadované referenční poloze. Ukázka vyznačení obálky hlavy ze snímku s odděleným pozadím je na obrázku 4-25.



Obrázek 4-25: Vyznačení obálky hlavy ve snímku s odděleným pozadím

### **Detekce náklonu řidiče doleva nebo doprava**

Díky informaci o referenční poloze hlavy je nyní možné začít vytvářet algoritmy pro detekci jednotlivých poloh řidiče. Všechna funkcionality detekce náklonu může být vytvořena v rámci jedné funkce – *MoveDetection*. Jako první se nabízí vytvoření detekce pro náklon řidiče doleva nebo doprava.

Náklon doleva či doprava lze charakterizovat změnou polohy hlavy ve směru osy x. To znamená, že se se změnou polohy hlavy budou měnit i souřadnice bodů kontury,

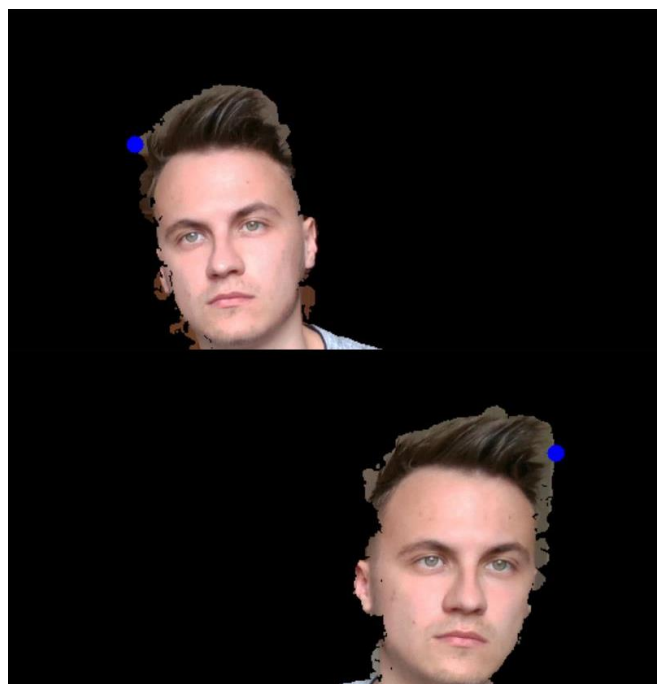
která ji popisuje. V rámci kontury ovšem není potřeba pracovat se všemi body. Pro každý příchozí snímek postačí najít body, které popisují krajní polohu hlavy z obou stran. Pro detekci náklonu doleva je potřeba pracovat s bodem kontury, kterého souřadnice  $x$  má nejmenší hodnotu ze všech bodů. Analogicky pro detekci doprava lze využít bod s největší hodnotou souřadnice  $x$ . Rychlost algoritmu se díky tomu zvýší, a to bez ztráty robustnosti.



Obrázek 4-26: Krajní body pro detekci náklonu do stran

Cílem je pochopitelně srovnat tyto krajní body s referenční polohou. Totožný postup zvolení bodů je tím pádem nutno aplikovat i na referenční konturu popisující základní polohu hlavy.

Na základě těchto znalostí bude funkce *MoveDetection* pracovat následovně: Nejprve nalezne obálku hlavy stejným způsobem, jako ve funkci pro změření referenční polohy hlavy. Poté musí v kontuře, což je dvourozměrné pole bodů se souřadnicemi  $X$  a  $Y$  najít krajní body obálky hlavy. To samé udělá i v rámci referenční kontury (může být provedeno i ve funkci *DetectionSetupScene1*). Následně je již možné jednotlivé body porovnávat a určovat rozdíly mezi aktuální a referenční polohou. Pro srovnávání je potřeba stanovit prahy jak pro levou, tak pravou stranu. Ty lze určit buď absolutně, ve formě limitních hodnot souřadnic, nebo relativně. Relativní způsob si lze představit jako procentuální hodnotu ve formě koeficientů, kterými se vynásobí referenční hodnoty krajních bodů. Tento způsob určení prahů je rozhodně lepší možnost, jelikož se umí přizpůsobit různé velikosti hlavy řidiče i různé poloze hlavy při referenčním měření. Alternativně lze v rámci projekční geometrie měřit odchylky bodů a určit prahy v metrech. Náklon hlavy doleva i doprava je vyobrazen na obrázku 4-27 níže.



Obrázek 4-27: Změna polohy krajních bodů při náklonu doleva a doprava

### **Detekce náklonu řidiče dopředu**

Další z pohybů, který má za cíl funkce *MoveDetection* detekovat, je náklon řidiče dopředu. Náklon dopředu teoreticky nepotřebuje data z měřicí funkce, ale při zpětném pohledu bude vhodné implementovat funkcionalitu detekce náklonu dopředu i do měřicí funkce. Je to z důvodu zamezení nechtěného náklonu dopředu při referenčním měření.

Pro detekci se i zde využije dat, které popisují aktuální polohu hlavy. V tomto případě se ale nevyžije krajních bodů, nýbrž hodnot bodů z hloubkového obrazu. Platí podmínka: Pokud má některý z bodů, který popisuje obálku hlavy, hodnotu v hloubkovém obrazu menší, než stanovený práh, řidič se naklání dopředu. Pro funkci to znamená, že musí najít minimum ze všech bodů v hloubkovém obrazu, které mají souřadnice shodné s body tvořící okraj hlavy. Následně hodnotu minima porovná se zvoleným prahem. Prah lze zvolit v metrech, je ovšem potřeba, stejně jako u funkce pro segmentaci celého řidiče, tuto hodnotu převést na jednotky používané v hloubkovém obrazu.

### **Detekce náklonu řidiče dovrchu**

I detekci náklonu řidiče dovrchu je možno vytvořit pomocí informace, kterou je možné obdržet popsáním obálky hlavy. Princip detekce je velice podobný detekci náklonu do stran, náklon dovrchu je ale detekovatelný na základě souřadnic typu  $Y$ . Je to pochopitelně proto, že souřadnice  $Y$  určují výšku zvoleného bodu v obrazu. Podmínka náklonu dovrchu zní takto: Pokud je souřadnice  $Y$  jakéhokoliv bodu, který popisuje



obálku hlavy menší, než zvolený práh, řidič se naklání dovrchu. Z toho vyplývá, že v rámci funkce *MoveDetection* postačí najít „nejvyšší“ bod tvořící okraj hlavy a jeho souřadnici *Y* srovnávat s prahovou hodnotou.

Pro tento algoritmus je rovněž možné zvolit buď absolutní, nebo relativní hodnotu prahu. Z důvodu zmíněných výhod se i zde zvolí relativní hodnota prahu. Aby ji bylo možné vypočítat, je nevyhnutelné stejnou funkcionalitu nalezení nejvyššího bodu hlavy přidat i do měřicí funkce *DetectionSetupScene1*. Poté je již možné zvolit koeficient, který vytvoří práh přizpůsobující se výšce každého řidiče.

## Zabezpečení proti falešným detekcím

Navržené algoritmy pro detekci jednotlivých typů náklonů řidiče ve vozidle jsou ve své podstatě velice jednoduché, jejich jednoduchost, bohužel, vytváří prostor pro falešné detekce.

První problém, který může nastat, se týká náklonu řidiče dovrchu. Pokud se řidič zvedne dovrchu, nebo postačí, že se v průběhu jízdy více narovná, kontura popisující největší objekt v horní polovině obrazu obkreslí i jeho ramena. To představuje velký problém pro algoritmus náklonu do stran. V takovém případě se jako krajní body (modrá barva) určí body, které se nachází na okraji obou ramen. Algoritmus detekce tím pádem bude hlásit relativně velkou odchylku, která může přesáhnout práh 10%, hlava ovšem může být v obvyklé poloze. Tuto situaci ukazuje obrázek 4-28.

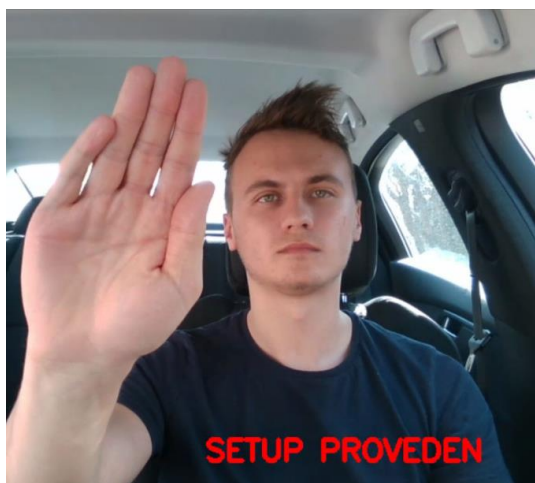


Obrázek 4-28: Detekce ramen při nadzvednutí

Vyřešit tuto problematickou situaci není složité. Původní algoritmus hledá oblast hlavy v horní polovině snímané plochy. Pokud se řidič nadzvedne, je nutné výšku analyzované oblasti proporčně zmenšit, a to právě o tolik, o kolik se řidič nadzvedne. To znamená, že se oblast zmenší o hodnotu rozdílu mezi referenční a aktuální hodnotou

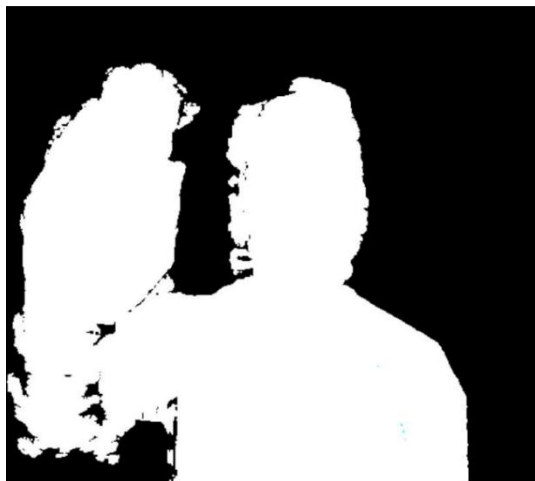
nejvyššího bodu hlavy. Po implementování tohoto principu do funkce *MoveDetection* již k nevhodné detekci ramen nebude docházet.

Druhá komplikace, která způsobí falešnou detekci náklonu, je zvedání rukou řidiče. Pokud má vozidlo řidiče takovou úroveň automatizace, že nemusí držet ruce na volantu, může se zároveň naklánět a tím pádem zde není co řešit. Pro úroveň automatizace 0 až 2 se ale řidič naklánět nesmí. Na druhou stranu, pokud drží volant alespoň jednou z rukou, druhou z nich zvedat může. Algoritmy náklonu pracují pouze s horní polovinou snímané scény, pokud řidič pohybuje rukou v dolní polovině, k falešné detekci nedojde. Problematická je horní polovina. Jestliže se ruka dostane do jedné z limitních poloh buďto pro detekci náklonu do stran, dopředu nebo dovrchu, může k falešné detekci dojít. Faktem je, že se tak nestane vždy, pouze v případě, že je kontura popisující ruku větší, než ta, která popisuje hlavu. Aby bylo možné falešné detekci tohoto typu zabránit, je tím pádem důležité rozeznat, zdali největší kontura detekovala hlavu nebo jiný objekt.



Obrázek 4-29: Problematické zvednutí ruky

Rozeznat detekci hlavy od detekce jiných objektů nemusí být příliš snadné. Cesta k řešení bude spočívat jednak ve využití dat z měřicí funkce, tak v charakteristických vlastnostech hlavy. Co se týče dat z měřicí funkce, zde se nabízí určit průměrnou plochu v pixlech, kterou řidičova hlava zabírá. Při následném porovnávání s detekovanou plochou v aktuálním snímku lze očekávat, že v případě detekce ruky bude plocha mnohem menší, než když se bude jednat o detekci hlavy. Tohle ale nemusí platit vždy. Čím blíže bude řidičova ruka ke kameře, tím větší bude její plocha v pixelech. U některých poloh tím pádem hrozí, že bude velikost plochy ruky velice podobná referenční velikosti hlavy. Částečně by šlo tento nedostatek zmírnit počítáním plochy na základě vzdáleností jednotlivých bodů v metrech, čímž by se odstranil problém „zvětšování“ ruky při přibližování se ke kameře. I v takovém případě by ale bylo možné dostat ruku do takové pozice, kterou by program detekoval jako hlavu.



Obrázek 4-30: Problematické zvednutí ruky - plocha

Myšlenku měření plochy hlavy je potřeba zdokonalit. Další možností, jak odlišit hlavu od ruky je předpoklad, že se v případě falešné detekce nachází hlava řidiče zhruba na podobné pozici, jako při prvotním měření. V rámci funkce *DetectionSetupScene1* je možné obdržet tzv. krajní body hlavy, které byly rozebrány u algoritmů pro detekci náklonu do stran a dovrchu. Tyto lze pak použít pro vytvoření okna (výřez snímané plochy), ve kterém je možné výskyt hlavy při povoleném chování řidiče očekávat. Výřez se určí na základě souřadnic bodů zvětšených o přiměřenou délku tak, aby šlo toto okno prohlásit za bezpečnou oblast, ve které se řidičova hlava může pohybovat, aniž by daná poloha hlavy byla prohlášena za náklon do strany nebo dovrchu (směr pohybu po ose  $X$ , resp.  $Y$ ). Ukázka takového okna je na obrázku níže.



Obrázek 4-31: Bezpečné okno

V rámci tohoto okna je nyní možné kontrolovat, zdali se v něm nachází analyzovaná kontura a jaká je velikost plochy, kterou popisuje. Může zde nastat několik situací. V prvním případě se jedná o situaci, kde je možné konturu v oknu nalézt a velikost plochy je podobná referenční velikosti. V tomto případě se zcela jistě jedná o hlavu řidiče. Další typ pohybu, který se může vyskytnout, je v případě mírného náklonu řidiče dopředu (pohyb ve směru osy  $z$ ). V takovém případě je hlava pořád v okně a může mít velikost větší než referenční. Platí, že u obou z těchto situací je hlava v okně a tudíž je povoleno zvedat ruku bez toho, aby systémy hlásily jakýkoliv náklon.

Na druhou stranu, pokud se v okně kontura nenajde vůbec nebo je popsána plocha značně menší, než referenční velikost, znamená to, že se hlava řidiče začíná naklánět směrem k nepovoleným pozicím. Pro celý program to zároveň musí znamenat, že řidič již nesmí zvedat ani jednu z rukou. Tato podmínka se dá obhájit potřebou zvýšené pozornosti řidiče z důvodu potenciálně nebezpečných náklonů. Je také nutná pro správnou funkčnost algoritmů a zachování možnosti zvedat ruce při bezpečné poloze. Jinak by se mohlo stát, že by algoritmus v případě polohy hlavy v jedné z limitních poloh a zároveň zvednuté ruky nemusel zvolit „správný“ objekt, který má hodnotit.

Speciální případ se týká zvednutí hlavy řidiče dovrchu. Referenční okno zůstává neměnné, při zvedání se ovšem v okně objeví oblast ramen a trupu řidiče. Hodnota nalezené plochy v okně může být klidně větší, je proto vhodné při dosažení horní části referenčního okna zakázat zvedání rukou. Díky tomu nebude hrozit falešná detekce náklonu dovrchu v případě zvednutí ruky nad hlavu.

Navržené řešení pro zamezení falešných detekcí při zvednutí rukou lze v rámci programu implementovat bez toho, aby bylo potřeba doplňovat funkci *MoveDetection*. Pro změření plochy hlavy je nejlepší vytvořit novou funkci s názvem *HeadPositionArea*. Tato funkce si nejprve vytvoří zmiňovaný výřez – okno, ve kterém následně hledá největší konturu. Poté co ji nalezne, spočítá pomocí knihovny funkce *contourArea* velikost obkreslené plochy. Obdobně je potřeba spočítat velikost plochy hlavy i ve funkci pro referenční data. Podmínku přítomnosti hlavy v okně lze napsat v hlavním bloku programu, kde musí platit podmínka, že pokud je velikost alespoň o trochu menší, než referenční velikost hlavy nebo byla dosažena horní hranice referenčního okna, řidič se dostává do nebezpečné polohy a nelze zvedat ruce. To bude signalizovat věta „**Řidič nesmí zvedat ruce**“. V opačném případě je jejich zvedání možné a program nevypíše oznámení o žádném z náklonů.

## Testování detekce jednotlivých náklonů řidiče (měřicí scéna č. 1)

Pro otestování funkcionalit, které mají za cíl detekovat jednotlivé možné náklony hlavy řidiče, bylo v prvním kroku nutné stanovit příslušné prahy. Ty byly určeny tak, aby popisovaly polohy hlavy řidiče, které citelně zhoršují možnost řidiče sledovat provoz a věnovat se řízení. V rámci testování programu je ovšem možnost tyto prahy před startem programu určit podle potřeby. Nutno také dodat, že celý proces testování z důvodu bezpečnosti probíhal v odstaveném vozidle.

Pro detekci náklonu do stran byl zvolen relativní práh s hodnotou koeficientu 1,1. To znamená, že pokud se řidič vychýlí ze své referenční polohy doprava o 10 %, resp. doleva o -10 %, funkce zaznamená pohyb do jedné ze stran. U detekce náklonu dopředu se jednalo o práh, který popisoval vzdálenost 0,45 m. Určení prahu v reálné aplikaci by ale muselo být dynamické, závisí totiž od nastavení sedadla. Řidič velkého vzrůstu by si

dozajista posunul sedadlo úplně dozadu, tím pádem by hodnota prahu pro detekci jeho náklonu dopředu musela být větší, než u menšího řidiče. Tuto informaci by musel poskytovat nějaký snímač, který by algoritmu posílal data o aktuální poloze sedadla. Nakonec, pro detekci náklonu dovrchu byla zvolena hodnota koeficientu 1,2.

Další konstantu, kterou bylo nutné v rámci hlavního programu stanovit, byl čas, po který je řidiči povoleno setrvat v některém z nebezpečných náklonů. To dává řidiči jistou míru volnosti, např. pokud se potřebuje nějakým způsobem poškrábat nebo protáhnout. V rámci náklonů je rozumná hodnota třeba 2 sekundy. Jelikož snímání probíhalo rychlostí 30 fps, hodnota dvou sekund představuje 60 snímků.

Po určení prahů a časové konstanty bylo možné přistoupit k otestování navržených algoritmů. To si vyžadovalo provést po spuštění programu referenční měření. Řidič v takovém případě musel sedět ve vzpřímené poloze po dobu několika snímků (desetiny vteřiny). Poté byly k dispozici data pro porovnávání s jednotlivými algoritmy náklonu. V rámci testování referenčního měření byly vyzkoušeny i chybové polohy, konkrétně držení hlavy mimo středové polohy a náklon hlavy dopředu. V takovém případě program měření neprovedl a vyzval řidiče pro návrat do korektní polohy.

Po provedení referenčního měření byl jako první z algoritmů otestován detektor náklonu do stran. Ten fungoval spolehlivě a vždycky při náklonu hlavy o více než 10 % zahlásil chybnou polohu hlavy. Zároveň při vychýlení hlavy z „bezpečného“ okna došlo k detekci nebezpečného stavu a program zakázal zvedání rukou.



Obrázek 4-32: Testování náklonu hlavy do stran

Problém mohl nastat při zvednutí řidiče dovrchu. Samotná detekce náklonu dovrchu fungovala bezchybně. Při zvednutí řidiče došlo k proporčnímu zmenšení horní poloviny snímku. Algoritmus zaznamenal hodnotu nejmenší souřadnice Y a při hodnotě menší než práh správně detekoval náklon dovrchu. Zároveň neumožnil falešnou detekci do stran, jelikož nedetekoval ramena.

Na druhou stranu, pokud se řidič zvednul do vrchu a zároveň mírně doleva, objevil se problém, který způsobovalo neúplné odstranění pozadí. Konkrétně část mezi A a B sloupkem vozidla, která byla nalezena při testování funkce pro segmentaci na základě hloubky. Problém spočíval v tom, že v případě, že se hlava dostala do oblasti neodděleného pozadí, algoritmus tuto oblast považoval za jeden celek. Limitní hodnoty kontury se tím pádem změnily a již nemusely popisovat krajní body hlavy. Tuto komplikaci lze v rámci navržených řešení algoritmů řešit jenom obtížně, smysl by dávalo ověření polohy očí nebo úst a následné rozhodnutí, jestli se jedná o falešnou detekci nebo nikoliv. Jelikož není detekce náklonu do vrchu až tak nebezpečná, oproti náklonům do stran a dopředu, není ve výsledném programu aktivní.

Detekce náklonu, která plně využila výhody stereo kamery, byla detekce náklonu dopředu. Ta se může vyskytnout, když například řidič usíná a jeho hlava se začne naklánět. I zde nebyl s funkčností algoritmu problém, při překročení prahu ve směru osy z došlo k detekci náklonu dopředu. Zajímavější situace nastala v případě, že byl řidič nakloněn dopředu a zároveň do jedné ze stran. Při přiblížení se ke kameře se logicky zvětšila zobrazená plocha hlavy. Při použití relativních prahů se kvůli tomu hlava lehce přiblížila k limitní hodnotě i bez toho, aby došlo k značnému náklonu do strany.

Tuto komplikaci lze řešit pouze přepočtem hodnot bodů, které tvoří konturu hlavy na hodnoty v metrech. Otázka je, zdali to má smysl. Platí totiž, že čím blíže ke kameře, tím kratší je skutečná šířka plochy, která se na kameře zobrazí a ve které je možné se pohybovat. Při přepočtu by se tím pádem mohlo stát, že pro blízké polohy hlavy vůči kameře by se práh pro detekci náklonu do stran dostal mimo zorné pole kamery. Lze konstatovat, že primárně je důležitá informace o náklonu dopředu. Náklon do stran má smysl určovat především v případech, že se řidič nachází v obvyklé, vzpřímené poloze. Příklad náklonu dopředu, který by mohl způsobit i detekci náklonu do strany je na obrázku níže:

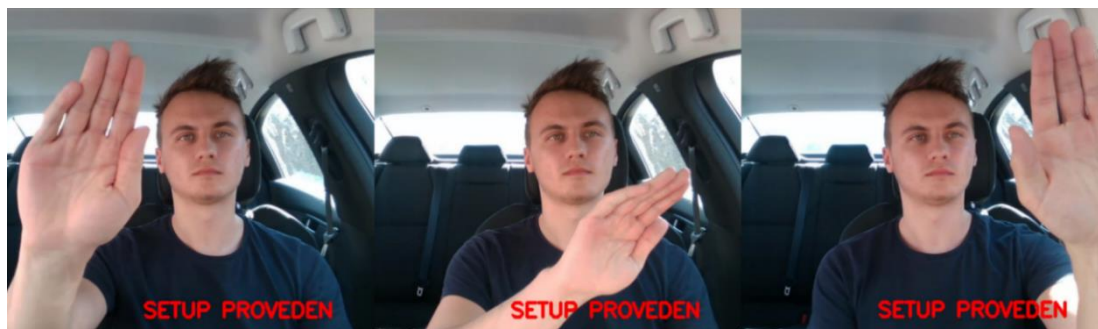


Obrázek 4-33: Testování náklonu hlavy dopředu



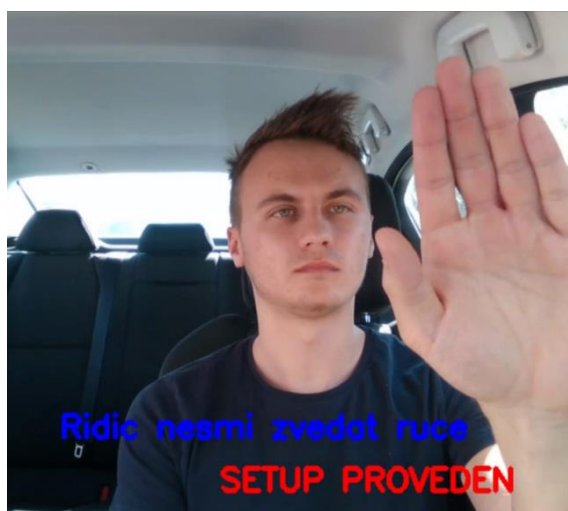
Celkově je možné zhodnotit, že navržené algoritmy náklonu při testování standardních situací pracují podle očekávání. Detekce má své nedostatky v případě špatně odděleného pozadí.

Další testování, které bylo potřeba provést, se týkalo situací, kdy řidič zvednul jednu z rukou bez toho, aby se naklonil do některé ze stran. V takovém případě program nesměl detekovat náklon. I v tomto případě se předpoklady naplnily. Program díky hlídání polohy hlavy v referenčním oknu nebral ohled na zvednuté ruce, a tudíž zcela správně nedetekoval náklon. Ukázky je možné vidět na obrázku 4-34.



Obrázek 4-34: Testování odolnosti vůči falešným detekcím

I přes dobrou funkčnost není ochrana proti falešným detekcím dokonalá. Problém nastává u jedné specifické situace, kdy se obrys ruky „spojí“ s hlavou a spolu vyvolá dosažení horního limitu v rámci referenčního okna. V takovém případě sice nedojde k falešné detekci náklonu dovrchu, program ale zakáže zvedat ruce. Všechny ukázky je možné zhlédnout ve formě videa s názvem ***Sledování řidiče | Matuš Pieger | Diplomová práce***. Odkaz na video se nachází v příloze této práce.



Obrázek 4-35: Ukázka chybně vyhodnocené situace při zvednutí ruky

#### 4.3.4 Detekce držení volantu řidičem

Dalším z algoritmů, který je potřeba navrhnout a zrealizovat je algoritmus detekce držení volantu, který se uplatní u systému se stejným názvem. Princip detekce, který by mohl spolehlivě fungovat, byl naznačen již v základním rozboru v podkapitole 4.1. Spočívá v porovnávání velikosti segmentované plochy s referenční hodnotou. U 2D snímků byla segmentace rukou celkem obtížná a závisela od kontrastu s pozadím. Při použití stereo kamery se situace i u této měřicí scény značně zjednodušuje. Pro segmentaci lze použít vytvořenou funkci, která je schopna segmentovat obraz na základě jeho hloubky. U volby prahu je ovšem nutné zamyslet se nad vzdáleností jednotlivých prvků, které se v měřicí scéně nachází. Kamera snímající volant není s plochou volantu úplně rovnoběžná, spodní část volantu je ke kameře blíže, než vrchní část. Při položení rukou na vrchní část volantu by se při nevhodně zvoleném prahu mohlo stát, že funkce bude tuto část rukou považovat za pozadí. Aby se tomuto stavu předešlo, je nejvýhodnější zvolit práh, který pokryje celý volant, zároveň ale neobsáhne oblast přístrojové desky nebo dalších součástí interiéru. Takový práh je možné snadno najít, jelikož zmíněné oblasti jsou od volantu vzdálené celkem daleko. Zároveň je možné konstatovat, že plocha volantu ve výstupním snímku nijak nepřekáží. Je to jednak z toho důvodu, že se jedná o objekt v neměnné vzdálenosti od kamery, ale také z důvodu navrhovaného principu detekce. Jako referenční snímek tedy poslouží snímek s vysegmentovaným volantem. K vzájemnému porovnání pak poslouží aktuální snímky, ve kterých se bude hlídat, zdali se v něm nachází i řidičovy ruce.

Po zavolání funkce s různými hodnotami prahů se při daném postavení kamery nakonec nejvíce osvědčilo použít hodnotu o velikosti zhruba 0,7 m. Taková hodnota spolehlivě oddělí volant i ruce na něm od pozadí, viz obrázek 4-36.



Obrázek 4-36: Oddělení pozadí u měřicí scény č. 2



Aby bylo možné porovnání s referenčním snímkem, vyplatí se i pro tuto měřicí scénu vytvořit funkci pro vstupní data, tentokrát s názvem *DetectionSetupScene2*. Tato funkce podobně, jako měřicí funkce pro první scénu pracuje s obrázkem s odděleným pozadím. Jedná se o obrázek s volantem bez rukou, to znamená, že při referenčním měření nesmí mít řidič ruce na volantu. Pro výpočet referenční hodnoty plochy volantu je následně potřeba pro několik prvních snímků spočítat počet pixelů s nenulovou hodnotou. Po zprůměrování se obdrží referenční hodnota plochy volantu bez rukou. Stejný výpočet je poté potřeba provádět i pro aktuálně přichozí snímky, tento výpočet je proto vhodné dát do samostatné funkce s názvem *GetSceneArea*.

Po obdržení potřebných dat je potřeba pro detekci držení volantu učinit již pouze poslední krok. Ten spočívá v určení relativního prahu, který řekne, o kolik musí být aktuální hodnota plochy větší, než referenční hodnota, aby bylo možné prohlásit, že se na volantu nachází ruce řidiče. Tato otázka souvisí i s podmínkou, jestli musí mít řidič na volantu obě ruce, nebo postačí pouze jedna z nich. Při podmínce obou rukou by musela být hodnota prahu o něco větší, než u podmínky jedné ruky. Držení volantu jednou rukou je běžný jev, který nijak významně nezhoršuje bezpečnost provozu vozidla, tím pádem postačí zvolit práh, který detekuje již menší zvětšení plochy segmentovaného snímku.

## Testování detekce držení volantu (měřicí scéna č. 2)

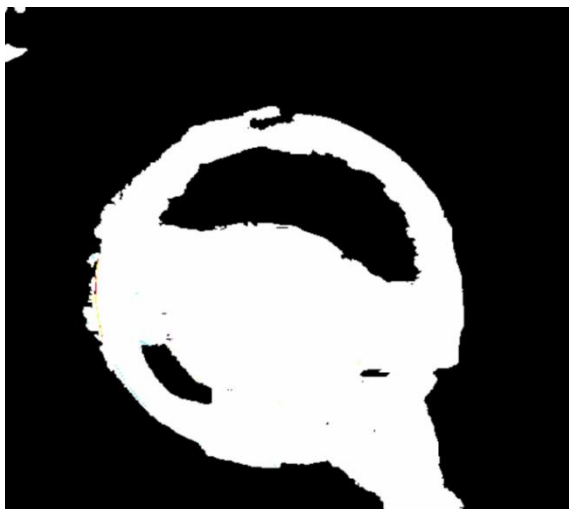
Při černobílém náhledu na vysegmentované plochy je očividné, že v případě držení volantu oběma rukama nebude detekce velký problém.



Obrázek 4-37: Ukázka segmentovaných ploch u měřicí scény č. 2

Pro určení prahu, který pokryje i situace, kdy řidič drží volant pouze jednou rukou, je vhodné zamyslet se, která pozice ruky na volantu dává nejmenší rozdíl hodnot vysegmentovaných ploch. Platí, že čím výše se ruce nachází, tím větší plochu zaberou.

Nejmenší rozdíl hodnot, nejspíše vznikne v situaci, kdy řidič drží volant někde v jeho dolní části.



Obrázek 4-38: Limitní situace při držení volantu jednou rukou

V této limitní situaci je možné naměřit zvětšení segmentované plochy o cca 10 %. Vhodnou hodnotu prahu může představovat třeba koeficient 1,08, resp. zvětšení o 8 %. Co se týče časové konstanty, po kterou je možné setrvat v nebezpečném stavu, u této měřicí scény ji není potřeba řešit. Vyplývá to z toho, že řidič má při držení volantu možnost držet ho pouze jednou rukou.

Po spuštění programu pro otestování detekce držení volantu bylo i u této měřicí scény potřeba provést referenční měření. V reálné aplikaci by ale toto měření nebylo nutné. Jelikož je celá oblast volantu statická, referenční data by šlo naměřit ještě v továrně a uložit je při výrobě auta do paměti. Tím pádem není potřeba vymýšlet, jak zamezit situacím, kdy by řidič při měření nerespektoval doporučení nechat ruce mimo dohled kamery.

Samotná detekce byla velice spolehlivá a nevykazovala víceméně žádnou chybovost. Algoritmus nijak neovlivnily změny světelných podmínek a lze ho považovat za poměrně robustní, což je do jisté míry zásluha stereo kamery. Příklady detekce jsou na obrázku 4-39. Další ukázky se nachází ve zmíněném videu.



Obrázek 4-39: Testování detekce držení volantu

### 4.3.5 Detekce připravenosti stlačit pedály

V rámci měřicí scény č. 3 je cílem navrhnout algoritmus, který bude schopen detekovat připravenost nohou řidiče stlačit pedály. Na základě úspěchu metody, která byla schopna spolehlivě detekovat držení volantu, je možné obdobný princip aplikovat i zde.

V základním rozboru bylo definováno, že o bezpečné poloze nohou lze mluvit tehdy, je-li alespoň jejich část viditelná ve snímané ploše. Jinak řečeno to znamená, že jsou nohy k pedálům dostatečně blízko a řidič je připraven, v případě potřeby, ihned zasáhnout. Pro rozlišení situace, kdy jsou nohy v bezpečné poloze, je opět možné postupovat na základě porovnávání s referenčním snímkem. Ten bude představovat takový obraz, ve kterém se nohy vůbec nenachází. Pro obdržení referenčních dat poslouží funkce *DetectionSetupScene3*. Tato funkce bude pracovat téměř úplně stejně jako měřicí funkce pro detekci držení volantu. Algoritmus detekce nohou je ale možné, za předpokladu, že řidič nohy nekříží, koncipovat tak, že bude detekovat levou a pravou nohu zvlášť. Pro změření referenční plochy levé či pravé nohy se obraz s odděleným pozadím vertikálně rozdělí na dvě poloviny. Poté se změří plocha vysegmentovaných oblastí pro obě poloviny zvlášť.

Samozřejmě pro účely této funkce je nevyhnutelné, stejně jako u měřicí scény č. 2, nejprve správně oddělit nohy, případně další objekty od pozadí. I zde platí, že není možné najít takový práh, který by spolehlivě segmentoval pouze oblasti nohou, aniž by se ve vzdálenosti menší než práh, nevyskytovaly i jiné objekty. V případě měřicí scény č. 3 se jedná o oblast plastového krytu, který se nachází pod volantem. Rovněž se do výstupního snímku po zvolení vhodného prahu dostanou pedály. Jelikož se jedná o statické objekty, nevadí to. Prah, který byl zvolen pro testování, představoval hodnotu 0,45 m. Při takové hodnotě je možné obdržet následující snímky:



Obrázek 4-40: Oddělení pozadí u měřicí scény č. 3

Po obdržení aktuálního snímku se pokaždé provede stejný výpočet vysegmentované plochy, jako u měřicí funkce. Nakonec je potřeba u obou polovin obrazu změřit odchylku od referenční plochy. Pokud není hodnota větší, než referenční hodnota zvětšená o koeficient, detektor určí, že daná noha není připravena zasáhnout.

### Testování detekce připravenosti stlačit pedály (měřicí scéna č. 3)



Obrázek 4-41: Ukázka segmentovaných ploch u měřicí scény č. 3

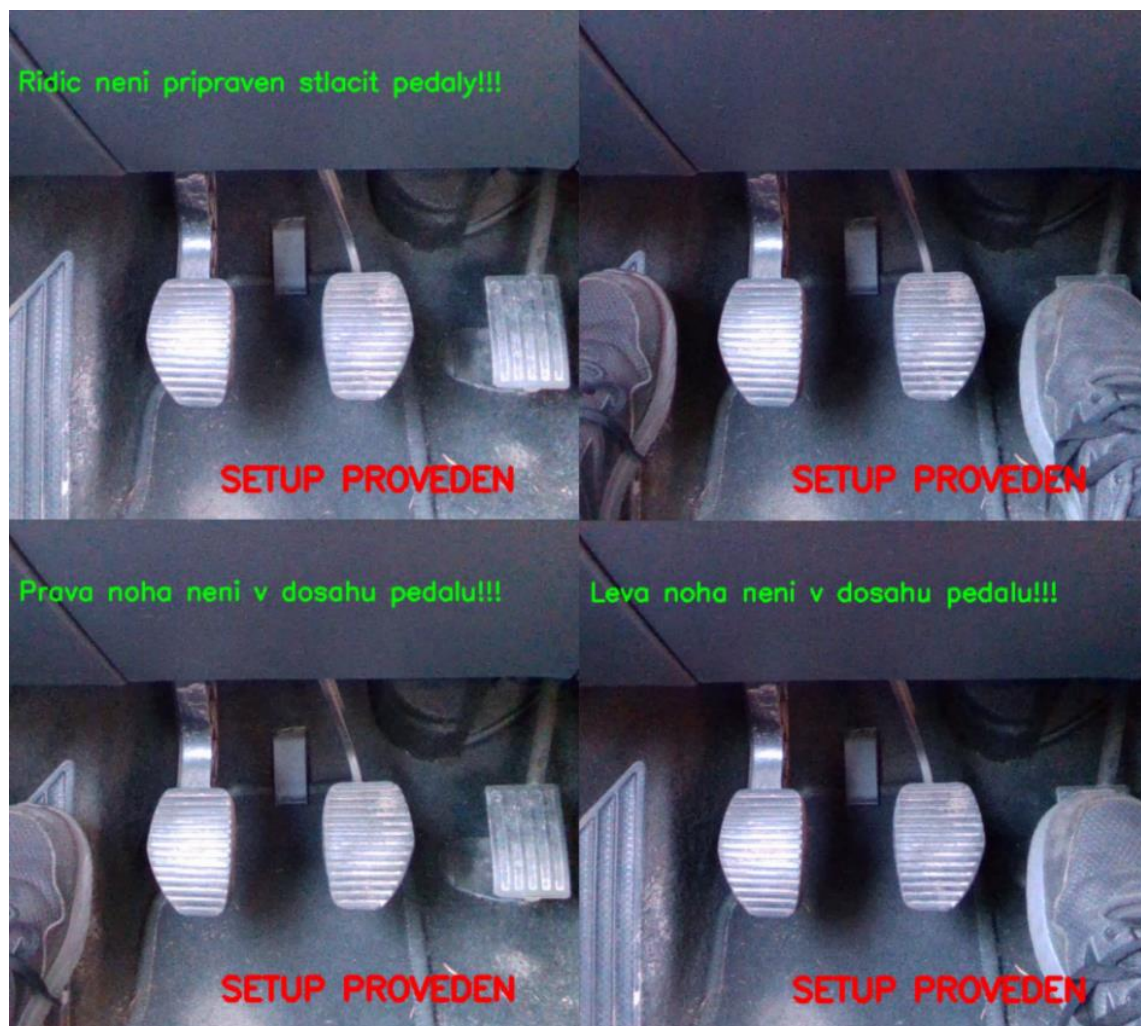
Oddělené plochy jsou vyobrazeny v černobílém provedení výše. Velký rozdíl ve vyobrazených plochách je v případě, že jsou nohy blízko pedálů. Při hledání vhodného prahu je opět potřeba najít práh, který pokryje i situace, kde je rozdíl v plochách menší. Jedná se o situace, kdy se nohy nenachází přímo na pedálech, jako je tomu u obrázku výše, ale pořád se nachází ve snímané ploše. Limitní polohu nohy, kterou lze ještě pořád považovat za bezpečnou, může představovat obrázek 4-42.



Obrázek 4-42: Limitní situace polohy nohy k pedálům

Při změření rozdílu ploch u limitní polohy nohy je možné naměřit zvětšení o cca 20 %. Prahový koeficient tedy bude mít hodnotu 1,2 a lze ho zvolit pro obě poloviny obrazu stejně. Časová konstanta zde stejně jako u měřicí scény č. 2 nebude využita.

Po určení prahové hodnoty bylo možné spustit program a provést referenční měření. Ani u této scény by měření nemuselo v reálné aplikaci probíhat – využily by se data z továrního nastavení. Při testování detekce pracoval algoritmus bezchybně, dokázal rozlišit bezpečný stav, ale také stavy, kdy v dosahu pedálů chyběla jedna z nohou nebo obě. Příklady detekce jsou na obrázku níže a ve videu.



Obrázek 4-43: Testování detekce připravenosti řidiče stlačit pedály

#### 4.3.6 Detekce mrkání a zívání

Zcela jiný algoritmus, který poslouží k vytvoření systému pro hlídání ospalosti řidiče, spočívá v detekci délky mrkání a zívání řidiče. Detekce mrkání byla v rámci práce již vyzkoušena v podkapitole, která popisovala možné algoritmy pro dvourozměrné snímky. Po rešerši bylo zjištěno, že se v praxi u detekce z kamerových snímků používá víceméně výhradně algoritmů strojového učení. Vyzkoušená metoda spočívala v principu



Haarovy kaskády. Byla využita natrénovaná neuronová síť, která poměrně spolehlivě detekovala oblast očí i za měnících se světelných podmínkách, ovšem vyskytovaly se i falešné detekce. Po obdržení dat popisujících oblast očí bylo však potřeba vymyslet algoritmy, které by v dané oblasti odlišily stavy, kdy je oko otevřeno a kdy je zavřeno. Pro řešení bylo vyzkoušeno použití metody pro měření odchylek počtu hran a také metoda GMM. Obě metody nebyly dostatečně robustní, u metody GMM byl též problém ve výpočetní náročnosti.

Snímky ze stereo kamery v tomto případě nejspíše situaci významným způsobem nepomůžou. Jedna možnost využití souřadnic  $z$  se ale nabízí, a to pro detekci zívání. Při znalosti oblasti, ve které se ústa vyskytují, lze v dané oblasti naměřit v průměru větší hodnotu souřadnic  $z$  (vzdálenost od kamery), v případě, že má řidič ústa plně otevřená. Srovnání s referenční hodnotou (zavřené ústa) je nicméně velice obtížné. Je to z toho důvodu, že se řidič během jízdy může mírně pohybovat směrem dopředu nebo dozadu. Tím pádem je spíše potřeba hledat rozdíly v homogenitě oblasti úst. V případě zavřených úst lze očekávat, že všechny body oblasti budou o podobné hodnotě souřadnice  $z$ . U otevřených úst tomu tak nebude.

Po prvotním otestování návrhu, použitím neuronové sítě pro detekci úst, bylo ověřeno, že tato metoda může v omezené míře fungovat, rozdíly mezi rozlišovanými stavy jsou ale poměrně malé. Při náklonu hlavy do jedné ze stran nebo nedostatečném otevření úst se metoda stává nespolehlivou. Bylo také zjištěno, že při značném otevření úst přestává fungovat detekce řešená neuronovou sítí. I takto by do jisté míry šlo detekovat zívání, nelze to ale považovat za spolehlivou metodu. Detekci očí nelze na základě hloubkových souřadnic vytvořit, rozdíl v hloubce mezi otevřeným a zavřeným okem je minimální.

Pro detekci očí a úst bude nakonec nejlepší využít princip popsáný v podkapitole asistenčních systémů navázaných na řidiče. V ní byla rozebrána studie, která popisovala výpočet tzv. očního poměru. Jednalo se o výpočet, který bylo možné provést na základě znalosti polohy charakteristických bodů popisujících oko. Rozlišování mrkání spočívalo ve značném poklesu hodnoty poměru při zavření oka. Podobný princip by dozajista šlo aplikovat i na oblast úst. Pro určení těchto bodů bude potřebné využít neuronovou síť z knihovny dlib.

## **Knihovna dlib**

Knihovna dlib je jedna z knihoven, která obsahuje, podobně jako *OpenCV*, spoustu užitečných funkcí pro práci s obrazem. Oproti *OpenCV* obsahuje navíc funkci tzv. prediktoru tvarů, který umožňuje v analyzovaném snímku najít hledaný tvar. V našem případě je potřeba najít obrys hlavy a příslušné charakteristické body tváře, které tvoří právě oblast očí nebo úst. Predikce tvaru se provádí v již nalezeném okně, které obsahuje

obličeje, šlo by tudíž použít již vyzkoušený detektor obličeje pracující na principu Haarovy kaskády. Knihovna *dlib* ovšem pro detekci obličeje v real-time používá detektor založený na histogramu orientovaných gradientů a metodě podpůrných vektorů. Ten je možné použít díky funkci *get\_frontal\_face\_detector*. Na základě své podstaty by HOG detektor mohl fungovat o něco lépe, především by měl vykazovat menší míru falešné positivity.

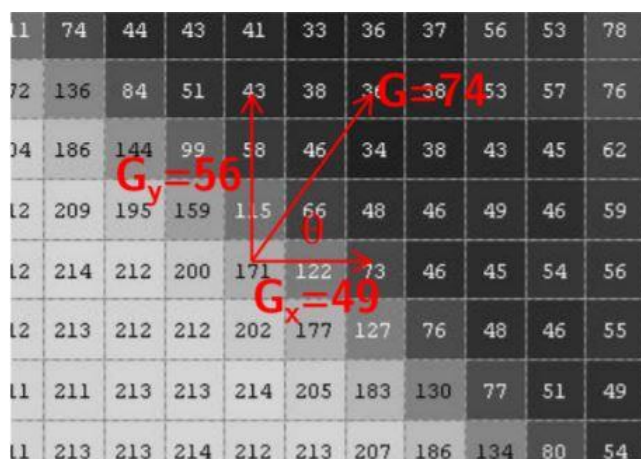
## Detekce obličeje na základě histogramu orientovaných gradientů

I když je dokumentace ke knihovně *dlib* jenom stručná, z obecných znalostí o principu HOG detektorů se lze dozvědět, že tato metoda detekce spočívá v několika krocích.

V prvním kroku obvykle dochází k předzpracování obrazu, které zahrnuje zmenšení obrazu a také gama korekci. Poté je možné přistoupit k výpočtu histogramu orientovaných gradientů. Výpočet se provede, jak pro směr v ose  $x$ , tak pro směr v ose  $y$ . Pro výpočet se použije např. Sobelův operátor. Pro obdržení celkové hodnoty gradientu v daném místě a jeho orientace se využije následujících vzorců [34]:

$$G[f(x, y)] = \sqrt{G_x^2 + G_y^2} \quad (8)$$

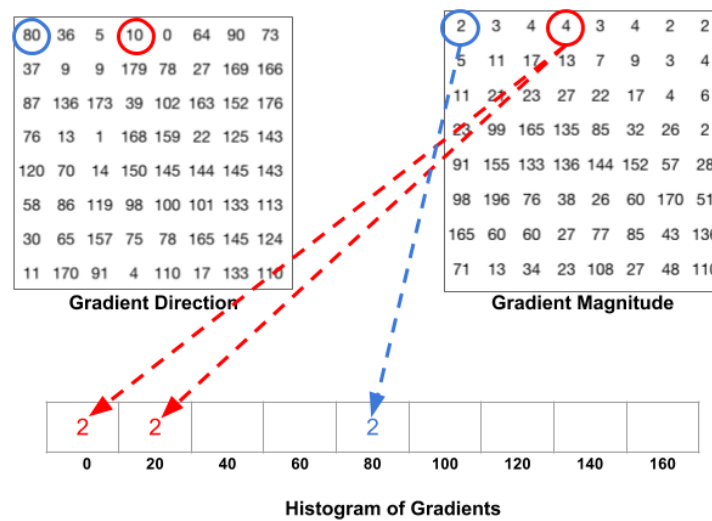
$$\theta(x, y) = \arg(G_x, G_y) \quad (9)$$



Obrázek 4-44: Výpočet gradientů v analyzovaném snímku [34]

V dalším kroku se analyzovaný snímek rozdělí na buňky o velikosti  $8 \times 8$ . V každé buňce je poté nutné vypočítat histogram nalezených gradientů. Samotný histogram je tvořen devíti „zásobníky“. Jedná se o devět hodnot úhlu natočení počínaje od 0 až po 160 stupňů s přírůstkem 20 stupňů pro každý následující zásobník. Při výpočtu histogramu se každá hodnota gradientu přičte do příslušného zásobníku podle toho, jakou hodnotu úhlu natočení daný gradient obsahuje. Pokud se hodnota nerovná žádnému ze

zásobníků, proporčně se rozdělí mezi dva nejbližší zásobníky. Grafickou reprezentaci výpočtu ukazuje obrázek 4-45.



Obrázek 4-45: Příklad výpočtu histogramu orientovaných gradientů [35]

Po obdržení HOG pro každou z buněk je ve třetím kroku potřebné vypočtené histogramy normalizovat. Je to pro odstranění vlivu různých světelných podmínek v obrazu. Metoda HOG pro normalizaci používá čtyři histogramy, které dohromady tvoří plochu 16x16. Tyto histogramy se převedou na vektor o velikosti 36 hodnot. Po provedení normalizace prvního vektoru se proces posune v obrazu o hodnotu 8 pixelů. To se opakuje až do doby, než jsou znormalizovány všechny vypočtené vektory histogramů.

V posledním kroku se tyto vektory sloučí do jednoho a vytvoří dlouhý vektor [35]. Ten popisuje celý analyzovaný snímek a je ho možné dále využít pro metodu podpůrných vektorů, což je metoda strojového učení s učitelem. Ta je schopna natrénovat model, který spolehlivě detekuje obličeje za různých světelných podmínek.

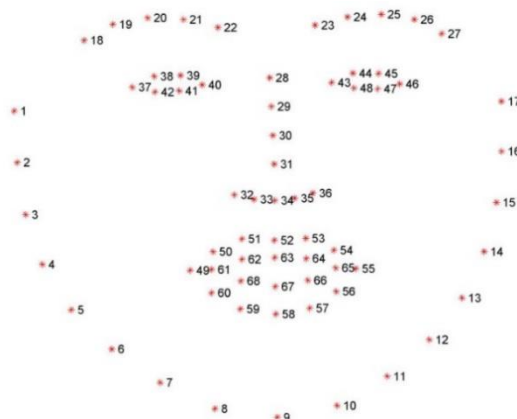
## Nalezení charakteristických bodů obličeje

Poté co je jasné, jak funguje metoda HOG pro nalezení oblasti obličeje je možné pustit se do využití hlavní výhody knihovny *dlib*. Ta, jak bylo zmíněno, spočívá v nalezení charakteristických bodů, které společně tvoří obličej řidiče. Tuto funkcionalitu poskytuje funkce *shape\_predictor*. Jako argument přijímá další natrénovanou neuronovou síť. V rámci knihovny *dlib* se jedná o síť s názvem *shape\_predictor\_68\_face\_landmarks*. Ta byla vytvořena na základě studie z roku 2014, na které se podíleli Vahid Kazemi a Josephine Sullivan [36]. Popisuje metodu, při které se nejprve u každého z obrázku patřícího do trénovacího datasetu manuálně označí charakteristické body obličeje. Také se zde pracuje s pravděpodobností vzdálenosti mezi páry vstupních pixelů. Pro detekci



se využívají tzv. regresní stromy, které jsou natrénovány tak, aby byly schopny odhadnout charakteristické body přímo z jasových hodnot pixelů.

Prediktor je schopen odhadnout tvar obličeje nalezením 68 charakteristických bodů, viz obrázek níže.



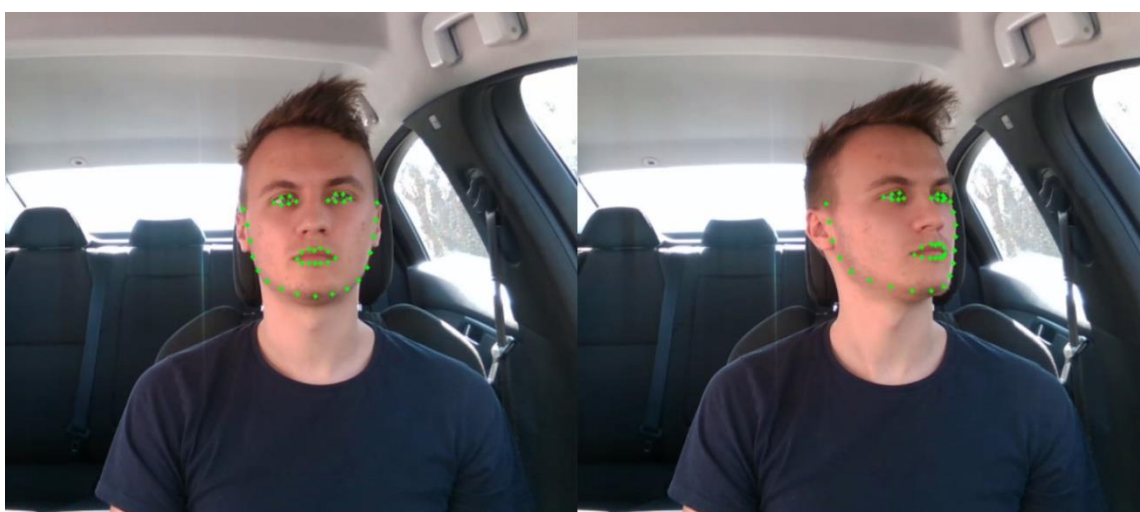
Obrázek 4-46: Charakteristické body obličeje prediktoru z knihovny *dlib* [37]

Pro detekci očí a úst jsou zajímavé především body 37-42, 43-48 a 49 až 60. Pro obdržení těchto bodů je dobré vytvořit novou funkci s názvem *FaceLandMarks*. Ta pracuje se snímkem s odděleným pozadím, zároveň si na základě funkce detektoru obličeje nalezne všechny obličeje, které se ve snímku nachází. V rámci měřicí scény č. 1 se jedná pouze o jeden obličej řidiče. V dalším kroku se využije prediktor (funkce *shape\_predictor*), který určí tvar obličeje nalezeného ve čtverci. Jak detektor, tak prediktor lze nainicializovat v hlavním programu a poté je ve funkci *FaceLandMarks* používat jako parametry.

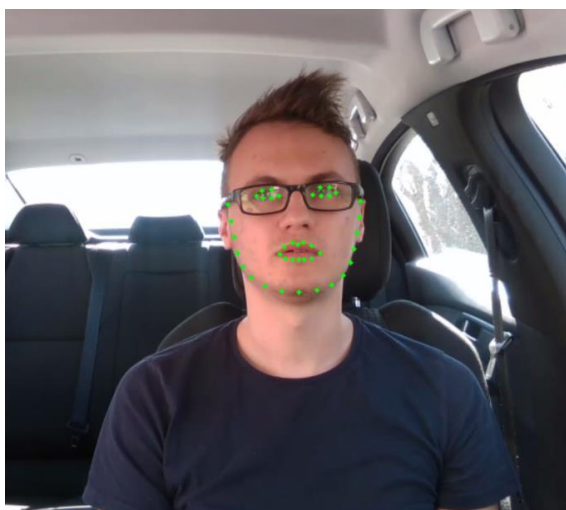
Pro další práci s nalezeným tvarem obličeje je dále vhodné vytvořit funkci *GetLandMarksCoordinates*, která extrahuje jednotlivé souřadnice charakteristických bodů do vhodné podoby. Funkce umožňuje filtrovat výběr souřadnic tak, že je možné extrahovat pouze ty, které jsou pro účely programu zajímavé. Jedná se tedy o oblast očí a úst. Poté co jsou souřadnice k dispozici lze vytvořit funkci, která bude počítat poměr stran očí, resp. úst. Jedná se o implementaci vzorce č. 1, který byl popsán v podkapitole asistenčních systémů s návazností na řidiče. Alternativně lze otázku mrkání a zívání řešit i výpočtem plochy, kterou charakteristické body označují, tuto funkcionalitu zabezpečuje funkce *GetLandMarksArea*, která vznikla jako prvotní nápad pro detekci mrkání. Funkce ovšem počítá plochu v pixelech, což není při náklonu řidiče ve směru osy Z vhodné. Nicméně při korekci naměřené plochy v závislosti na vzdálenosti od kamery by se tento způsob dal použít.

## Testování algoritmu pro detekci ospalosti (měřicí scéna č. 1)

Otestování funkcionality prediktoru tvaru obličeje pomocí charakteristických bodů potvrdilo předpoklad spolehlivosti algoritmu. V rámci testování proběhlo vykreslení oblasti očí, úst a čelisti. Algoritmus je možné hodnotit jako velice rychlý, což umožňuje použití v real-time, zároveň si umí poradit s různým natočením hlavy. Selhává pouze v případě značného ohybu hlavy, což jsou ovšem situace, které řeší navržené detektory náklonů. Otestovat bylo potřeba i situaci, kdy byl sledován řidič, který nosí brýle. Prvotní metoda Haarovy kaskády měla s brýlemi celkem velký problém a vykazovala značnou míru falešné nebo žádné detekce. Tento algoritmus si ovšem s brýlemi uměl poradit, viz obrázek 4-48.



Obrázek 4-47: Testování prediktoru tvaru obličeje



Obrázek 4-48: Testování prediktoru tvaru obličeje – řidič s brýlemi

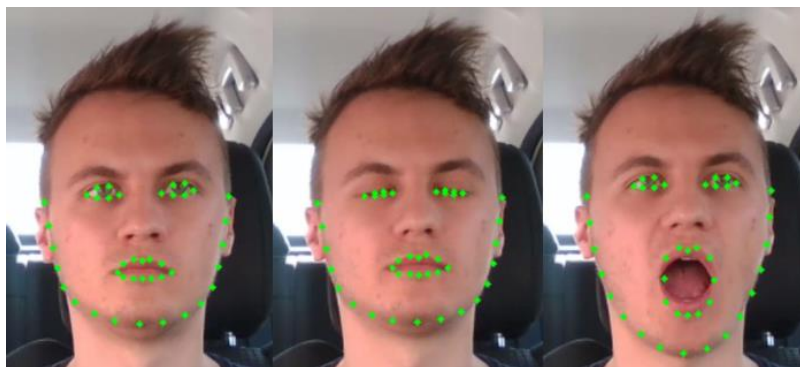
V kontextu detekce mrkání a zívání bylo opět potřeba určit prahové hodnoty. Mrkání je detekováno v případě, že je hodnota poměru menší, než práh (zavřené oči).

Tato hodnota při testování klesala téměř k nule, za zavřené oči šlo považovat situaci, kdy poměr poklesl pod hodnotu 0,15. V rámci testování bylo zároveň zjištěno, že poměr očí je při pohledu řidiče mimo kameru jenom o něco větší, než v případě zavření očí. Tento fakt lze využít pro vylepšení funkcionality detektoru. Detektor nemusí hlídat pouze ospalost řidiče, ale také sledování vozovky. Zamezí to situacím, kdy řidič neusíná, ale dívá se delší dobu, např. na telefon nebo infotainment vozidla. Pro pokrytí těchto situací byl nakonec zvolen práh o hodnotě 0,2. Platí, že díky principu metody nebyla, v případě mírných náklonů řidiče, nutná žádná korekce prahu.

U stanovení prahu pro detekci zívání bylo nutné najít takovou hodnotu prahu, u které šlo spolehlivě prohlásit, že má řidič plně otevřená ústa. Podmínka plně otevřených úst byla potřebná z důvodu zamezení falešné detekce zívání, v případě, že řidič mluví. Osvědčilo se nastavit práh poměru úst na hodnotu 0,7.

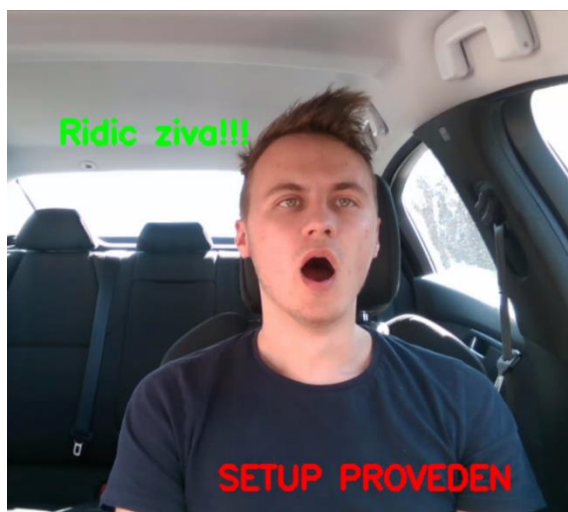
Po úspěšném určení vhodných prahů zbývalo zamyslet se nad časovými konstantami, které se aplikují na jednotlivé subsystémy detektoru. Algoritmus usínání nově funguje i jako hlídač pozornosti řidiče. Ta je velice důležitá, již sekunda nepozornosti je schopna vyvolat nebezpečnou situaci na silnici. Jako dobrý kompromis se jeví časová konstanta o velikosti 0,5 s, resp. 15 snímků. U detekce zívání je zase vhodné zamyslet se, jaká je průměrná délka, po kterou jsou ústa člověka při zívání plně otevřena. Běžně se jedná o délku několika sekund, dobrým prahem pro vyvolání detekce zívání je hodnota zhruba 1,5 sekundy.

Při použití zmíněných konstant byla detekce mrkání-pozornosti i zívání poměrně spolehlivá. Problematické fungování vykazovala pouze při větších náklonech hlavy. Algoritmus také nefunguje při značném natočení hlavy do jedné ze stran. To je pochopitelné, jelikož za takových podmínek není oblast očí a úst plně viditelná. V rámci programu se proto detekce pozornosti a zívání deaktivuje, pokud program detekuje některý ze sledovaných náklonů nebo natočení hlavy, což je funkcionality, kterou bude ještě potřeba dotvořit. Podmínka deaktivace souvisí s celkovým provázáním jednotlivých algoritmů v programu a bude rozebrána blíže v poslední kapitole práce. Na obrázku 4-49 je ukázáno, jak se zmenší plocha očí po zavření, resp. jak se po otevření zvětší plocha úst.



Obrázek 4-49: Ukázka změny nalezených ploch při zavření očí a při zívání

Obrázky 4-50 a 4-51 ukazují samotnou funkčnost algoritmu pro detekci zívání a usínání - pozornosti.



Obrázek 4-50: Testování detekce zívání



Obrázek 4-51: Testování detekce usínání, resp. pozornosti řidiče

### Vylepšení zabezpečení proti falešným detekcím na základě obrysu hlavy

Díky nalezení spolehlivé metody pro detekci tvaru obličeje dává smysl vrátit se k problematice falešných detekcí při sledování jednotlivých náklonů. Ta spočívala v hlídání přítomnosti hlavy v tzv. bezpečném okně, ve kterém se může hlava řidiče nacházet, aniž by došlo k detekci některé z nepovolených pozic. Toto hlídání lze vylepšit použitím dat, které popisují nalezený tvar obličeje. Pokud se, například obrys čelisti oddálí ze své referenční polohy o zvolenou délku, lze určit vychýlení z bezpečné polohy. Nicméně, pro účel detekce tohoto vychýlení není potřeba analyzovat všechny charakteristické body čelisti. Postačí se zaměřit na středový bod (v mapě charakteristických bodů se jedná o bod č. 9) a u něj měřit odchylky od referenční polohy. Obdobně by šlo použít některý z bodů pro popis očí, úst nebo dokonce nosu.

V rámci implementace algoritmu se referenční poloha bodu, resp. jeho souřadnice  $x$  a  $y$  nejprve změří v měřicí funkci *DetectionSetupScene1*. Poté je možné formulovat dvě podmínky: 1. Pokud je aktuální poloha ve směru osy  $x$  větší než zvolený práh, řidič se dostal mimo bezpečné okno a hrozí náklon doleva nebo doprava. 2. Pokud je aktuální poloha ve směru osy  $y$  větší, než zvolený práh, řidič se dostal mimo bezpečné okno a hrozí náklon dovrchu nebo dopředu. I když se spojitost souřadnice  $y$  s náklonem dopředu může jevit jako nelogická, platí, že při tomto typu náklonu dojde i k poklesu výšky hlavy. Obě podmínky lze zahrnout do hlavní části programu. Toto řešení může fungovat samostatně, ale i jako doplňkové řešení prvotní metody popsané v podkapitole *Zabezpečení proti falešným detekcím*.

### 4.3.7 Detekce telefonování

Na základě předešlých znalostí je dále možné vytvořit algoritmus pro detekci telefonování. Dá se postupovat dvěma způsoby.

#### Použití principu ploch

První princip detekce, který se nabízí je velice jednoduchý a známý. Jedná se o detekci na základě rozdílů velikosti segmentovaných ploch. Algoritmus tedy pracuje stejně, jako je tomu u detekce držení volantu a detekce připravenosti stlačit pedály. Pro hledání odchylky v plochách lze použít již vytvořené okno obsahující pouze hlavu řidiče. Nalezení tohoto okna s hlavou bylo rozebráno v podkapitole, která řešila zamezení falešným detekcím.

Telefonování bude detekováno v případě, že řidič přiloží svou rukou telefon k uchu. Vysegmentovaná plocha se tím pádem zvětší, viz obrázek 4-52. Zároveň platí, že se řidičova hlava musí nacházet v „bezpečném“ okně, což je podmínka, která je akceptovatelná. V případě odchýlení hlavy od povolené pozice program stejně detekuje některý z náklonů.



Obrázek 4-52: Ukázka zvětšení měřené plochy pro detekci telefonování

Toto řešení může u obvyklých situací fungovat, procentuální změna zvětšení plochy ovšem nemusí být moc velká a závisí od velikosti ruky i samotného úchopu telefonu. Při mírném náklonu řidiče dopředu nejspíše nebude možné odlišit, zdali je zvětšení plochy od referenční hodnoty způsobeno náklonem, nebo telefonováním.

## Detekce telefonování na základě měření skutečné šířky hlavy

Pro odstranění falešných detekcí, které mohou vznikat při mírném náklonu řidiče dopředu, je nutné přepočítat souřadnice kontury hlavy do jednotek vzdálenosti. Pro tento účel je potřebné znát zákonitosti projekční geometrie. Ta popisuje zobrazování snímaných bodů na rovinu obrazového snímače, jedná se tedy o zobrazení z 3D do 2D. Vztah, který popisuje projekční model, a tudíž převod mezi souřadnicemi v pixelech na souřadnice v metrech vypadá následovně [38]:

$$x = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & s & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} * \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = K(I|0) * X \quad (10)$$

Jedná se o model s ohniskem v počátku systému souřadnic, což je případ kamery Intel RealSense. V případě, že by tomu tak nebylo, dostala by se do hry matice vnějších parametrů kamery.

Souřadnice v pixelech jsou značeny malými písmeny  $x$  a  $y$ . Souřadnice v metrech jsou naopak označeny velkými písmeny  $X$ ,  $Y$  a  $Z$ . Jako neznámé vystupují hodnoty  $X$  a  $Y$ , hodnotu souřadnice  $Z$  v metrech je možné obdržet z hloubkové mapy. Pro převod souřadnic je dále potřebné znát hodnoty matice vnitřních parametrů  $K$ . Konkrétně hodnoty ohniskové vzdálenosti a střed projekce kamery. Hodnoty je možné najít ve struktuře *rs2\_intrinsics*.

Po obdržení všech potřebných hodnot je možné přistoupit k přepočtu. Ten lze provést funkcí *rs2\_deproject\_pixel\_to\_point*. Funkcionalitu přepočtu je nejlepší dát do nové funkce, např. s názvem *GetRealDistance*. Pro měření šířky hlavy je vhodné využít krajní body kontury, funkci lze proto navrhnout tak, aby vracela reálné hodnoty těchto dvou bodů najednou. Poté je možné vytvořit funkci *HeadSize*, která vrátí samotnou šířku hlavy v metrech. Ta se vypočte jako rozdíl hodnot dvou krajních bodů. Pro umožnění porovnávání s referenční šířkou musí být tato funkce použita i v měřicí funkci *DetectionSetupScene1*.

Na základě známé šířky hlavy v metrech lze nyní říct, že telefonování je možné detekovat, pokud se šířka hlavy v důsledku přiložení telefonu k uchu zvětší o prahovou hodnotu. Tato hodnota bude zároveň invariantní vůči náklonům řidiče dopředu. Tuto podmínku je dobré dát do nové funkce s názvem *CallDetection*.



## Testování detekce telefonování (měřicí scéna č. 1)

Pro otestování algoritmu bylo v prvním kroku, jako obvykle, potřebné určit potřebné hodnoty prahů. Zde se, v případě první metody, jednalo o určení, o kolik se musí plocha hlavy zvětšit, aby bylo možné prohlásit, že řidič telefonuje. Při prvotním měření bylo zjištěno, že se musí jednat alespoň o zvětšení o hodnotě 10 %.

U druhé metody bylo potřeba určit, o kolik se při přiložení telefonu k uchu zvětší šířka hlavy v metrech. Najít vhodný práh bylo u této metody o dost složitější. Tato hodnota je totiž hodně závislá na typu úchopu telefonu. Někdo si telefon přitlačí úplně k uchu, jiný si od ucha nechá menší odstup. To pochopitelně ovlivňuje naměřenou délku. Možné úchopy telefonu jsou na obrázku níže.



Obrázek 4-53: Možné úchopy při telefonování

Měřením bylo zjištěno, že nejmenší zvětšení při přiložení telefonu k uchu činí cca 2 cm, v závislosti od úchopu to ovšem může být dokonce 5-6 cm. Prah lze tudíž stanovit spíše jako interval hodnot, ve kterých je možné očekávat, že řidič telefonuje.

Po stanovení zmíněných prahů zbývalo určit vhodnou časovou konstantu. Při telefonování je předpoklad, že bude řidič setrvávat v dané poloze delší dobu. Je tedy možné zvolit hodnotu třeba 5 s. Tato hodnota zároveň spolehlivě zamezí falešným detekcím v situacích, kdy se řidič z různých důvodů dotýká hlavy, např. pro poškrábání se.

Po stanovení konstant mohlo proběhnout, jako obvykle, referenční měření. U první metody se potvrdil předpoklad náchylnosti metody na mírné náklony dopředu. V takových situacích algoritmus detekoval telefonování, i když řidič netelefonoval. Rovněž docházelo k falešné detekci při natočení hlavy do strany.

Druhá metoda byla již o něco spolehlivější, falešná detekce se při náklonu dopředu nevyskytovala. Při natočení hlavy ovšem i tato metoda vykazovala částečnou chybovost. Problematiku natočení hlavy bude potřeba vyřešit dalším algoritmem. Ukázky detekce telefonování jsou na obrázku 4-54.



Obrázek 4-54: Testování detekce telefonování

#### 4.3.8 Detekce natočení hlavy do strany

Poslední funkcionalita, kterou zbývá navrhnout je detekce natočení hlavy do strany. Tím se myslí otočení hlavy do jedné ze stran bez toho, aby se řidič naklonil. Pro detekci natočení by šlo využít funkci z předešlé podkapitoly pro přepočítání souřadnic z pixelů na metry. Vychází se z myšlenky, že při natočení hlavy bude možné naměřit větší hodnotu šířky hlavy. Neznamená to samozřejmě, že se řidičova hlava nějakým způsobem zvětší, využívá se faktu, že šířka hlavy z profilu je větší, než při měření šířky v přední části.

Nicméně při prvotním změření bylo zjištěno, že šířka profilu hlavy je příliš podobná hodnotám, které naměří algoritmus detekce telefonování. Tyto dva jevy tím pádem nejde příliš odlišit.

Po zvážení různých faktorů je možné zvažovat několik možností detekce. První možnost je vytáhnout z prediktoru tvaru obličeje informaci o poloze špičky nosu a čelisti. Při natočení hlavy doleva nebo doprava je možné očekávat, že se vzdálenost špičky nosu od kamery o něco zvětší. Vzdálenost se ale může zvětšit i při celkovém posunutí hlavy řidiče více dozadu. Právě proto je potřeba tuto informaci propojit s informací o vzdálenosti čelisti. Platí, že při větším natočení hlavy budou body popisující čelist blíže ke kameře, než špička nosu. Na druhou stranu prediktor tvaru při výrazném natočení hlavy selhává. Tuto metodu tím pádem lze použít pouze omezeně.

Tento fakt lze ovšem paradoxně proměnit ve výhodu. Podmínku natočení lze formulovat takto: Pokud není možné najít tvar obličeje, zároveň se ale hlava nachází v bezpečném okně (což lze zjistit z nalezené kontury hlavy), je možné říct, že je hlava natočena do jedné ze stran. V programu je tuto funkcionalitu možné přidat do funkce *FaceLandMarks*, samotnou podmínku pak do hlavní části programu.

Poslední alternativou je sledovat změny v poměru vzdálenosti čelisti a krku od kamery. Bylo by ale nutné najít spolehlivou metodu pro detekci krku.



## Testování detekce natočení hlavy (měřicí scéna č. 1)

Detekce natočení hlavy se testovala velice jednoduše, nebylo potřebné stanovovat žádný práh, také zde nemělo smysl zavádět časovou konstantu. Spolehlivost detekce byla velmi dobrá. Na druhou stranu, při natáčení hlavy doleva nebo doprava prediktor tvaru selhával až při velkém natočení. Za normálních okolností to je dobrá zpráva, zde se však jedná o nevýhodu. Souvisí to s algoritmem pro detekci telefonování, který nefunguje příliš spolehlivě při mírných natočeních hlavy. Vzniká tedy jistá „mezera“, kdy při určitém úhlu natočení není program schopen spolehlivě detekovat telefonování a zároveň neumí detekovat natočení hlavy.

Testovaný algoritmus podobně spolehlivě funguje i při natočení hlavy dolu nebo dovrchu. I v těchto případech je potřeba značný úhel natočení. Navíc může kvůli povaze pohybu dojít k aktivaci zabezpečení proti falešným detekcím. Příklady detekce natočení jsou na obrázku 4-55.



Obrázek 4-55: Testování algoritmu pro detekci natočení hlavy do stran

### 4.3.9 Výsledný program a provázanost algoritmů

Práce využívá tři různé měřicí scény, výsledný program tím pádem musí obsahovat tři podprogramy. Každý z nich využije pouze algoritmy, které obsluhují systémy využitelné pro danou měřicí scénu.

Podprogram č. 1 slouží pro snímání těla řidiče. Uplatní se v něm nejvíce navržených algoritmů a společně tvoří systém detekce připravenosti převzít řízení a systém detekce ospalosti, resp. pozornosti řidiče. Oba systémy běží v nekonečné smyčce nezávisle na sobě, v případě detekce některého z náklonů se ovšem deaktivuje systém hlídání ospalosti a také algoritmus pro detekci telefonování. To má dva důvody. Je to jednak proto, že tyto algoritmy spolehlivě fungují pouze v případě, že je viditelná

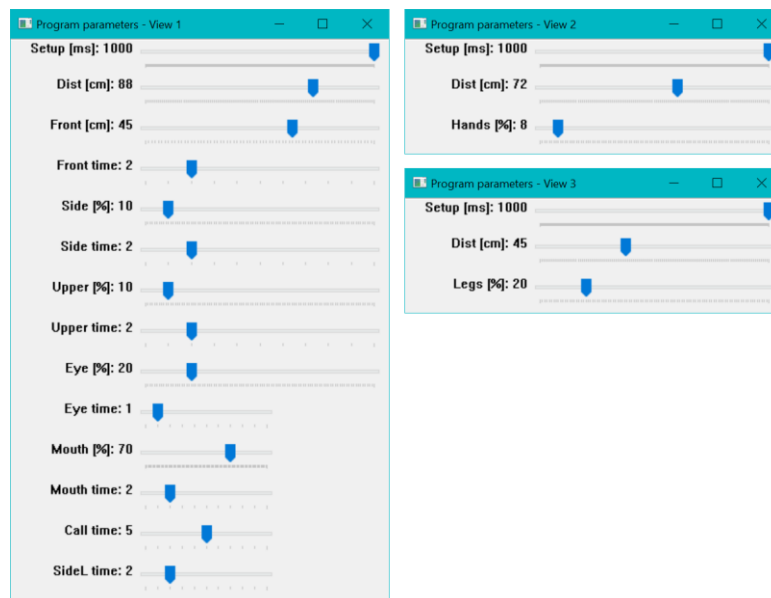
celá přední část obličeje řidiče. Na druhou stranu, už při detekci některého z náklonů se spustí varování před nebezpečím, další varování proto nemá moc smysl.

Podprogram č. 2 je určen pro obsluhu systému pro detekci držení volantu. Tento program by v případě reálné implementace do vozidla musel být propojen s podprogramem č. 1. Souvisí to se zamezením falešných detekcí při hlídání náklonů řidiče. V případě opuštění bezpečného okna má řidič zakázáno zvedat jakoukoliv z rukou, podprogram č. 2 by tedy musel podprogramu č. 1 dodávat informaci o tom, jestli jsou obě ruce skutečně na volantu. Současné řešení poskytuje informaci pouze o tom, zdali je na volantu alespoň jedna ruka, algoritmus by proto bylo potřeba mírně předělat na způsob algoritmu detekce nohou, kde se každá noha hlídá zvlášť.

V rámci podprogramu č. 3 jsou volány funkce pro realizaci systému pro kontrolu umístění noh v blízkosti pedálů. U tohoto systému platí, že může fungovat zcela nezávisle na jiných systémech.

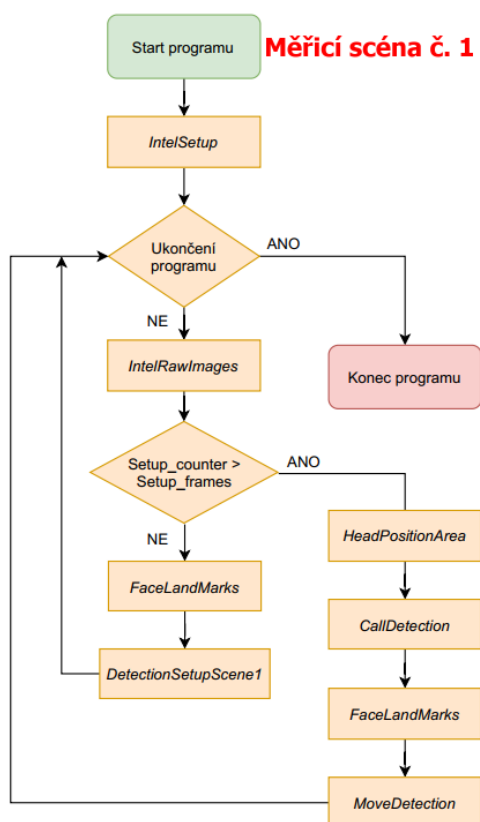
Jelikož byla v rámci realizace této práce k dispozici pouze jedna stereo kamera, nebylo možné otestovat všechny systémy pro sběr dat najednou. Všechny testy popsané v předešlých podkapitolách byly provedeny pro každou měřicí scénu zvlášť. Také platí, že se každý z podprogramů spouští samostatně.

Ke spuštění podprogramu pro konkrétní měřicí scénu postačí otevřít příslušný soubor. Ten nejprve vyzve uživatele k zadání potřebných prahových a časových konstant. Po navolení konstant je potřeba ukončit výběrové menu tlačítkem „q“. Poté, co je menu ukončeno je možné program plně používat. Pro správnou funkci programu je ovšem nutné mít připojenou kameru Intel RealSense D415 a také příslušné knihovny: *pyrealsense2*, *numpy*, *cv2*, *dlib*, *face\_utils* a *scipy*. Pokud chce uživatel pořídít kamerový záznam je k dispozici vytvořená funkce *IntelRecording* pro nahrávání videa ve formátu *.bag*.

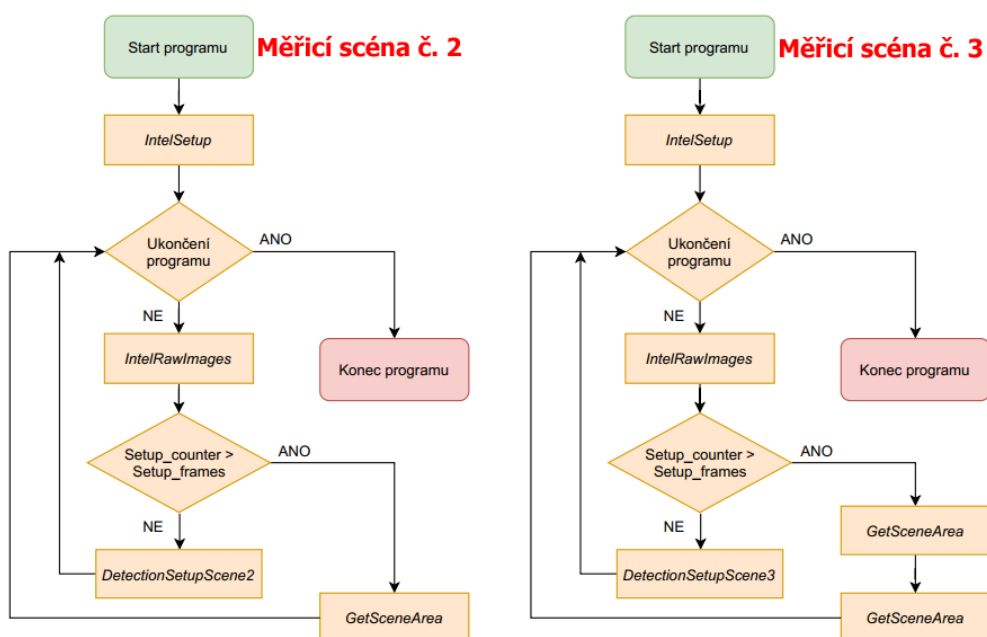


Obrázek 4-56: Ukázka menu pro zvolení potřebných konstant

Pro lepší pochopení struktury podprogramů se na obrázku níže nachází vývojové diagramy pro jednotlivé měřicí scény. Diagramy jsou zjednodušené a znázorňují pouze pořadí volání vytvořených funkcí.



Obrázek 4-57: Vývojový diagram – měřicí scéna č. 1



Obrázek 4-58: Vývojový diagram – měřicí scéna č. 2 a 3

## 5 ZHODNOCENÍ VYTVOŘENÝCH METOD PRO DETEKCI ŘIDIČE A JEHO POHYBŮ

Tato kapitola obsahuje celkové zhodnocení vytvořených metod, zamýšlí se nad nedostatky a navrhuje možná vylepšení.

Po otestování každého z vytvořených systémů je možné usoudit, že jednotlivé algoritmy fungují v obvyklých situacích, do kterých se řidič může dostat, celkem spolehlivě. Systémy tedy lze považovat za funkční prototypy, které by se mohly do jisté míry uplatnit v automobilech s nějakou formou autonomního řízení.

Samozřejmě možných situací a faktorů, které mohou správnou funkčnost systémů zmařit, je velké množství. Reálná implementace systémů tohoto typu by si dozajista vyžadovala mnohem větší míru testování. Jako autor práce jsem pro to bohužel neměl prostředky ani podmínky. Algoritmy se mi ale kromě sebe povedlo otestovat i na další osobě. Zajímavá by byla otázka chvění kamery při jízdě vozidla. Funkčnost algoritmů nebylo možné za jízdy vyzkoušet, lze ovšem odhadovat, že by se výraznější chvění kamer objevilo pouze při jejich nedostatečném upevnění ve vozidle. V takovém případě by bylo nejvhodnější srovnávat referenční hodnoty s průměrem hodnot z několika posledních snímků.

V rámci celkového zhodnocení se níže nachází tabulka kladů a záporů všech algoritmů z pohledu situací, ve kterých mohou selhat, resp. ve kterých fungovat určitě budou. V tabulce jsou také uvedeny nápady na odstranění popisovaných nedostatků.

### Detekce ospalosti řidiče na základě mrkání a zívání

---

- ✓ Funguje velmi spolehlivě při různých natočeních hlavy.
  - ✗ Při výrazném natočení algoritmus selhává.  
*Řešení -> Tyto polohy řeší ostatní algoritmy.*
  - ✗ Funkčnost realizovaného řešení se může zhoršit v případě horších světelných podmínek.  
*Řešení -> Použití kamery s IR přisvětlováním.*
- 

### Detekce připravenosti řidiče převzít řízení

---

- ✓ Princip segmentace na základě hloubky je víceméně bezchybný, nezávisí na světelných podmínkách.
- ✓ Algoritmy detekující jednotlivé náklony jsou velice jednoduché, v případě povedené segmentace nevykazují falešné detekce.

- ✓ Ochrana proti falešným detekcím funguje na většinu snímané plochy bez problémů.
  - ✓ Detekce natáčení hlavy do stran vykazuje dobré výsledky, pokud funguje funkce pro detekci ospalosti.
  - ✓ Telefonování se daří detekovat v případech, že je řidičova hlava v poloze podobné referenční.
  - ✗ Segmentace závisí na podmínce nepřítomnosti jiných objektů v podobné vzdálenosti od kamery jako tělo řidiče.  
*Řešení -> Položit kameru do takové polohy, ze které není možné nasnímat nežádoucí objekty – může být obtížně proveditelné.*
  - ✗ Ochrana proti falešným detekcím spouští zákaz zvedání rukou, v případě přílišného přiblížení se ruky nebo jiného objektu k hlavě řidiče, resp. k bezpečnému oknu.  
*Řešení -> Analyzování tvaru vzniklé kontury (?).  
Přímé rozeznávání ruky (?).*
  - ✗ Detekce telefonování není moc spolehlivá, při natočení hlavy do strany může docházet k falešné detekci. Při mírném vybočení z bezpečného okna, při kterém ještě nedojde k detekci některého z náklonů, může detekce selhat z důvodu špatného změření šířky hlavy.  
*Řešení -> Falešné detekce potlačit použitím informace o změně poměru oka při natočení hlavy (?) – může být nespolehlivé.  
Rozšířit bezpečné okno tak, aby se jeho hranice dotýkaly oblastí, ve kterých automaticky dojde k detekci některého z náklonů (?) – může způsobit jiné problémy.*
- 

### **Detekce držení volantu řidičem**

---

- ✓ Detekce porovnáváním aktuální velikosti oddělené plochy je velmi spolehlivá. Pokud nedojde k překrytí zorného pole kamery, detekce nemá důvod nefungovat.
- ✗ Falešné detekce se mohou vyskytovat při položení jiných objektů před volant.  
*Řešení -> Doplnková detekce držení volantu na kapacitním principu.*

### **Kontrola umístění noh v blízkosti pedálů**

---

- ✓ Stejný princip detekce jako u detekce držení volantu – bezproblémová funkčnost.
- ✗ I zde platí, že jiné předměty položené před pedály přispívají k falešným detekcím.  
*Řešení -> (?)*
- ✗ Při překřížení nohou selže rozeznávání mezi levou a pravou nohou.  
*Řešení -> (?)*

## 6 ZÁVĚR

Cílem diplomové práce bylo vytvořit systémy pro sběr dat, které hlídaly pozornost řidiče při řízení, resp. jeho připravenost převzít řízení v případě vozidel s vyšší úrovní automatizace.

Tento cíl se do velké míry povedlo naplnit. V rámci práce jsem jako autor vytvořil čtyři systémy využívající tři různé měřicí scény. První měřicí scéna byla zaměřena na pohled na řidiče, uplatnily se zde algoritmy hlídající náklon řidiče, resp. pohyby jeho hlavy do jednotlivých stran. Dále zde byla vytvořena funkcionality detektoru telefonování. Tyto algoritmy společně tvořily systém připravenosti převzít řízení. Vytvořenou detekci náklonů a ohybů do stran lze hodnotit jako spolehlivou a to především díky využití principu segmentace na základě hloubky v obrazu. To bylo umožněno díky zvolení stereo kamery pro obdržení zdrojových snímků. Detekce telefonování vykazovala menší nedostatky a nelze ji prozatím hodnotit jako dostatečně robustní.

V rámci pohledu na řidiče byl rovněž realizován systém pro hlídání pozornosti-ospalosti řidiče. Ten fungoval na základě hlídání velikosti oblastí očí a úst. Detekce těchto částí obličeje byla realizována nejnovější metodou pro určení tvaru obličeje a její spolehlivost lze hodnotit velice kladně. Oba systémy pro měřicí scénu č. 1 se hodí do vozidel s levellem automatizace až do hodnoty 4.

Měřicí scéna č. 2 zahrnovala pohled na volant a ruce řidiče. Systém, který tuto scénu využíval, dostal název detekce držení volantu řidičem. I tato detekce byla do velké míry možná na základě použití stereo kamery a spočívala v měření odchylek segmentovaných částí, tedy volantu a rukou. Systém se uplatní u vozidel s mírou automatizace 2 nebo nižší.

Záběr na nohy řidiče a pedály vozidla představovala měřicí scéna č. 3. Uplatnil se zde systém pro kontrolu umístění nohou v blízkosti pedálů. Tento systém využíval stejný princip jako algoritmus pro detekci držení volantu a může se využít u vozidel s levellem automatizace od 0 do 4. Systémy pro tyto měřicí scény je možné považovat za funkční, falešná detekce se zde při obvyklých situacích nevyskytovala.

Navržené systémy fungují společně jako jeden celek, videoukázky použití systémů je možné zhlédnout na videu, které se nachází v příloze této práce. Do budoucna je možné předpokládat, že se vývoj v oblasti automobilové bezpečnosti bude zaměřovat právě na systémy spojené s bezpečností autonomního řízení. V tomto kontextu lze hodnotit i přínos této práce. Ta na tuto problematiku poskytuje základní náhled a přináší řešení ve formě detekce rizikových situací, do kterých se řidič svými pohyby a chováním v případě nižších úrovní automatizace řízení (levely 0-2) nesmí dostat. V případě vyšších úrovní se detekce uplatní pro situace, kdy je potřeba, aby řidič převzal řízení.

## BIBLIOGRAFIE

- [1] *European Comission: Road fatalities per million inhabitants* [online]. [cit. 2020-12-22]. Dostupné z: [https://ec.europa.eu/transport/facts-fundings/scoreboard/compare/people/road-fatalities\\_en](https://ec.europa.eu/transport/facts-fundings/scoreboard/compare/people/road-fatalities_en)
- [2] NHTSA. *Traffic safety facts: Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey* [online]. Únor 2015 [cit. 2020-12-22]. Dostupné z: <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812115>
- [3] BLEAKNEY, Peter. *From headlights to seat belts: A brief history of automotive safety systems* [online]. 30.04.2019 [cit. 2020-12-22]. Dostupné z: <https://driving.ca/volvo/features/feature-story/from-headlights-to-seat-belts-a-brief-history-of-automotive-safety-systems>
- [4] *Injury Facts: Car Crash Deaths and Rates* [online]. [cit. 2020-12-22]. Dostupné z: <https://injuryfacts.nsc.org/motor-vehicle/historical-fatality-trends/deaths-and-rates/>
- [5] *Volvo seatbelt advertisement* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.caranddriver.com/news/a28775593/three-point-seatbelt-history/>
- [6] *Evolution of car safety features. The Automobile Association* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.theaa.com/breakdown-cover/advice/evolution-of-car-safety-features>
- [7] *Bosch cruise control* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.cartoq.com/every-car-in-india-to-have-automatic-braking-adaptive-cruise-control/>
- [8] *Night vision plus by Bosch. Autoevolution* [online]. 16. 11. 2009 [cit. 2020-12-22]. Dostupné z: <https://www.autoevolution.com/news/bosch-night-vision-plus-identifies-pedestrians-13270.html>
- [9] *Asistent pro jízdu v jízdním pruhu LKA (Lane Keeping Assist)* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.kia.com/cz/modely/sorento/objevte/>
- [10] *Ford Focus traffic sign recognition* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.prnewsprime.com/2019/06/traffic-sign-recognition-systems-market-research-report-2019-2025/>
- [11] *Example of pedestrian detection* [online]. [cit. 2020-12-22]. Dostupné z: <https://github.com/topics/oriented-gradients>
- [12] SYNOPSIS. *What is ADAS?* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.synopsys.com/automotive/what-is-adas.html>
- [13] THOMSON, Jamie a Florian RICHERT. *Autonomous driving: The 5 levels of autonomous driving explained* [online]. 09.03.2020 [cit. 2020-12-22]. Dostupné z: <https://www.intelligent-mobility-xperience.com/the-5-levels-of-autonomous-driving-explained-a-912861/>
- [14] SYNOPSIS. *The 6 Levels of Vehicle Autonomy Explained* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.synopsys.com/automotive/autonomous-driving-levels.html>

- [15] CDC. *Impaired Driving: Get the Facts* [online]. [cit. 2020-12-22]. Dostupné z: [https://www.cdc.gov/motorvehiclesafety/impaired\\_driving/impaired-driv\\_factsheet.html](https://www.cdc.gov/motorvehiclesafety/impaired_driving/impaired-driv_factsheet.html)
- [16] *Breath-Based Technology: System for alcohol measurement* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.dadss.org/breath-based-tech>
- [17] *Touch-Based Technology: System for alcohol measurement* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.dadss.org/touch-based-tech>
- [18] SOUKUPOVÁ, Tereza a Jan ČECH. *Real-Time Eye Blink Detection using Facial Landmarks* [online]. 2016 [cit. 2020-12-22]. Dostupné z: <http://vision.fe.uni-lj.si/cvww2016/proceedings/papers/05.pdf>  
Center for Machine Perception, Department of Cybernetics Faculty of Electrical Engineering, Czech Technical University in Prague.
- [19] HAUPT, Daniel, Petr HONZIK, Peter RASO a Ondrej HYNČICA. *Steering wheel motion analysis for detection of the driver's drowsiness* [online]. [cit. 2020-12-22]. Dostupné z: <http://www.wseas.us/e-library/conferences/2011/Tenerife/COMESDE/COMESDE-45.pdf>  
Department of Control and Instrumentation, Brno University of Technology.
- [20] KRAJEWSKI, Jarek, David SOMMER, Udo TRUTSCHEL, Dave EDWARDS a Martin GOLZ. *Steering Wheel Behavior Based Estimation of Fatigue* [online]. 2009 [cit. 2020-12-22]. Dostupné z: <https://ir.uiowa.edu/cgi/viewcontent.cgi?article=1311&context=drivingassessment>
- [21] SVENSSON, Ulrika. *Blink behaviour based drowsiness detection – method development and validation* [online]. Linköping, 2004 [cit. 2020-12-22]. Dostupné z: <https://www.diva-portal.org/smash/get/diva2:19913/FULLTEXT01.pdf>  
Linköpings tekniska högskola Institutionen för medicinsk teknik.
- [22] EOG. *Mindmedia* [online]. [cit. 2020-12-22]. Dostupné z: <https://www.mindmedia.com/en/solutions/research/eog/>
- [23] SINGH, Karamjeet a Rupinder KAUR. *Physical and Physiological Drowsiness Detection Methods* [online]. Punjab, India, 2013 [cit. 2020-12-22]. Dostupné z: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.403.4863&rep=rep1&type=pdf>
- [24] *Normal (colored) vs night-vision (black and white) view* [online]. [cit. 2020-12-22]. Dostupné z: <https://ecosportdiyseries.com/2018/08/04/dash-camera-with-night-vision-mode/>
- [25] Intel RealSense Depth Camera D415. *Intel Realsense* [online]. [cit. 2021-4-28]. Dostupné z: <https://www.intelrealsense.com/depth-camera-d415/>
- [26] Image Segmentation in Deep Learning: Methods and Applications. *Missinglink* [online]. [cit. 2020-12-22]. Dostupné z: <https://missinglink.ai/guides/computer-vision/image-segmentation-deep-learning-methods-applications/>
- [27] CARRASCO, Oscar Contreras. Gaussian Mixture Models Explained. *Towards data science* [online]. 03.06.2019 [cit. 2020-12-22]. Dostupné z: <https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95>



- [28] LEE, Soret. Understanding Face Detection with the Viola-Jones Object Detection Framework. *Towards data science* [online]. 22.05.2020 [cit. 2020-12-22]. Dostupné z: <https://towardsdatascience.com/understanding-face-detection-with-the-viola-jones-object-detection-framework-c55cc2a9da14>
- [29] MATAS, Jiří. Integrální obrázek. *Machine Vision Group* [online]. [cit. 2020-12-22]. Dostupné z: [http://vision.uamt.feec.vutbr.cz/ROZ/lectures/10\\_Detekce\\_objektu\\_pomoci\\_scanning\\_window.pdf](http://vision.uamt.feec.vutbr.cz/ROZ/lectures/10_Detekce_objektu_pomoci_scanning_window.pdf)
- [30] STARMER, Josh. AdaBoost, Clearly Explained. *StatQuest* [online]. 14.01.2019 [cit. 2020-12-22]. Dostupné z: <https://www.youtube.com/watch?v=LsK-xG1cLYA>
- [31] Canny Edge Detection. *OpenCV Documentation* [online]. [cit. 2020-12-22]. Dostupné z: [https://docs.opencv.org/master/da/d22/tutorial\\_py\\_canny.html](https://docs.opencv.org/master/da/d22/tutorial_py_canny.html)
- [32] Pasivní triangulace. *Machine Vision Group: Computer Vision* [online]. [cit. 2021-4-28]. Dostupné z: [http://vision.uamt.feec.vutbr.cz/POV/exercises/ex08\\_Pasivni%20triangulace.pdf](http://vision.uamt.feec.vutbr.cz/POV/exercises/ex08_Pasivni%20triangulace.pdf)
- [33] POUND, Mike. *Stereo 3D Vision - Computerphile* [online]. 24. 2. 2016 [cit. 2021-4-28]. Dostupné z: <https://www.youtube.com/watch?v=O7B2vCsTpC0>
- [34] HORÁK, Karel. Detekce hran a rohů. In: *Computer Vision Group* [online]. [cit. 2021-4-28]. Dostupné z: [http://vision.uamt.feec.vutbr.cz/ZVS/lectures/08\\_Detekce\\_hran\\_a\\_rohu.pdf](http://vision.uamt.feec.vutbr.cz/ZVS/lectures/08_Detekce_hran_a_rohu.pdf)
- [35] MALLICK, Satya. *Histogram of Oriented Gradients explained using OpenCV* [online]. 6. 12. 2016 [cit. 2021-4-28]. Dostupné z: <https://learnopencv.com/histogram-of-oriented-gradients/>
- [36] KAZEMI, Vahid a Josephine SULLIVAN. *One millisecond face alignment with an ensemble of regression trees* [online]. 2014 [cit. 2021-4-28]. Dostupné z: <https://ieeexplore.ieee.org/document/6909637>
- [37] Facial landmarks. *Datahacker* [online]. [cit. 2021-4-28]. Dostupné z: <http://datahacker.rs/009-how-to-detect-facial-landmarks-using-dlib-and-opencv/>
- [38] KLEČKA, Jan. Vícesnímková rekonstrukce scény. *Machine Vision Group: Recognition* [online]. [cit. 2021-4-28]. Dostupné z: [http://vision.uamt.feec.vutbr.cz/ROZ/lectures/08\\_Multisnimkova\\_rekonstrukce\\_objektu.pdf](http://vision.uamt.feec.vutbr.cz/ROZ/lectures/08_Multisnimkova_rekonstrukce_objektu.pdf)

## SEZNAM PŘÍLOH

Příloha této diplomové práce uložená na CD nosiči obsahuje následující složky:

1. Složku **program**, ve které se nachází zdrojové soubory vytvořených systémů.
2. Složku **video\_ukazky**, ve které se nachází textový soubor s odkazem na video na portálu YouTube. Video obsahuje ukázky funkčnosti všech navržených systémů.
3. Složku **diplomová\_práce\_xpiege00**, která obsahuje text práce ve formátu pdf.